AMATH 383 Final Project

Guang Hua, Shiqi Wang

University of Washington,

Seattle, WA 98195, USA

August 18, 2022

Abstract

We developed a mathematical way to reasonably predict the relationship between power consumption and performance of the graphic card, and we further introduced a new reference point for estimating the optimal operation range for getting a decent power efficiency. We calculated the model based on a single AMD Radeon RX 6600 XT graphic card and then generalized the model by introducing extra parameters that fit more situations. We will consider two main scenarios for this model: (1) Precise model for the RX 6600 XT graphic card, (2) Generalized model for graphic cards with RDNA 2 architecture (the same architecture RX 6600 XT is based on) The first model would be more accurate, and the second generic-generalized model will be more uncertain due to several factors we would like to neglect to simplify the problem. Building the Performance-to-power model could not only help potential buyers to estimate a power efficiency characteristic for the GPU product but also could potentially stimulate the graphic card market and manufacturers to further develop more new graphic cards that with higher performance but lower power consumption. This model could be further improved by adding consideration of power efficiency of graphic memory chips, power integrated circuit (IC) controller, logical board's power leaking, etc, which would be a topic more related to electrical engineering area that our team does not have adequate experiences with.

1 Group Member Tasks:

- Guang Hua: Main moderator. Focus: Idea brainstorming; Collecting data; Come up with basic model; research direction director; resource analyzer. Main area: Introduction, First basic model, generalized model, limitations, conclusion, plot & figures.

- Shiqi Wang: Main focus: Content search; Analyze data and elaborate idea; Looking for resources for research; Give feedback in physics area. Main area: Problem description, generalized model, conclusion.

2 Introduction

Human beings, as well as other primate animals, are highly visual creatures that explore surroundings and absorb new information every single second through visual senses [KB14]. Since the first computer was invented in 1945, the way that people gather information mainly switched from paper-based media to computers [Bur47]. The 21st Century is known as the era of information since nearly half of the global population owns at least one personal computer at home by 2019 [Als22]. People use computers for browsing the web, enjoying entertainment, doing office tasks, etc. The computer fetch information from the Internet and present them to the monitor screen to the user. In the process of compiling and representing the information into an understandable visual format, the graphic processing unit, short as GPU, is playing an essential component in the processing sequence.

The GPU is designed to accelerate image creation and processing of graphical effects and then output the result on a display device. Generally, there are two types of consumer GPU: integrated GPU and discrete GPU. Integrated GPU is designed for power-efficient devices like laptops and HTPCs, which are integrated inside the Central Processing Unit (CPU) or the motherboard. The discrete GPU, on the other hand, focuses more on high-performance areas such as gaming, image rendering, and training for artificial intelligence. The discrete GPU usually appears with a large PCB electrical circuit board with thermal solutions above the chip, which we generally call them the *graphic card*. the discrete GPU uses much more power than the integrated one, and sometimes even requires an external power supply besides the power delivery from the motherboard, but it could reach much higher peak performance and is designed to sustain a long period of high-demand computing [Mic].

In the current consumer GPU market, there are three mainstream manufacturers: Intel, NVIDIA, and AMD. Each manufacturer develops its specialized graphic architecture for its product. Intel, for example, uses Intel Xe (Gen 12) for its latest integrated graphic products, which aims for outstanding video encode/decode performance and artificial intelligence in image editing while maintaining a low-power specification [Int]. NVIDIA implements its Ampere architecture to achieve powerful real-time ray-tracing technology with artificial intelligence and machine learning [Nvi]. This project will mainly focus on the RDNA 2 architecture by AMD, which is a perfect example of balancing

performance and power efficiency. The RDNA 2 architecture first introduced hardware accelerated ray-tracing with the new RT Cores inside the compute Unit (CU). With the finer 7 nanometers manufacturing technology by TSMC and its architecture improvements, the new RDNA 2 architecture also shows up to 65 percent of performance per watt over the previous RDNA architecture and nearly 2.5 times uplifts over the old GCN architecture in the same area [AMDb].

3 Problem Description and Categorization

Modern electronic technologies take huge leaps every generation, containing faster processing capabilities and introducing newer features. As the GPU continues to evolve, there is one aspect that we should not ignore, though, which is the power efficiency of the processor during operation. A processor with high power efficiency could not only use less power while achieving similar performance compared to previous generations but also produce less heat emission during electric transmission loss, which potentially prolongs the system lifespan. Thus we decided to introduce a problem regarding the relationship between performance and power efficiency of the graphic processing unit during operation. To provide a deep understanding of the relationship between theoretical performance and power efficiency and give a well-defined model for this problem, we will categorize the problem into two distinct groups of samples. The first group is a single graphics card, the Radeon RX 6600 XT from AMD, which represents a specific balanced example as an RDNA 2 architecture product. We will use this card to collect essential data including various voltages, currents, and corresponding power draw from the power supply, which will help us to build a specific mathematical model based on the general physics formula of power consumption of semi-conductor. The second group contains all graphic cards with RDNA 2 architecture, which we will use to derive a generalized mathematical model for the RDNA 2 family by including specified parameters such as compute unit count and Video RAM capacity. This model will be an approximation since we could not determine whether each GPU may share the same characteristics without have deep understanding in hardware-level, despite they use the same RDNA 2 architecture. Instead, we will introduce other specialized parameters for each architecture as coefficients to approximate the power efficiency.

The reason we decided to analyze this problem is due to the high power consumption of recent graphic cards from the manufacturers above. One of the most representative examples is the GeForce RTX 3090 Ti from NVIDIA released on March 29th, 2022. As the most powerful product in the whole RTX-30 series line-up, the RTX 3090 Ti has the largest GPU processor die size that contains 10752 NVIDIA CUDA Cores and a whopping 1.86 GHz boost clock

frequency, which is 9.41 percent faster than the previous RTX 3090. From the benchmarks comparison between RTX 3090 Ti and RTX 3090 by TechSpot, there is an average 4 to 7 percent of performance increase in running 12 popular games under 1080p, 1440p, and 2160p resolution. In contrast, the power consumption increase is shocking: In running the game *Doom* with Ultra Nightmare Quality under 1440p resolution, the RTX 3090 achieved an average of 596 watts of system power load, while the RTX 3090 Ti, depends on different products from third-party vendors, superseded the previous record with the results of 673 watts and 702 watts, which is an average of 12.9 to 17.8 percent of power draw increase [Wal]. The higher power draw demand does not contribute much to the actual gaming performance in typical usage. To maximize the benefit of performance boost from power consumption, we decided to derive a mathematical model to approximate the best optimal operation range in the relationship between power consumption and performance output to achieve the best efficiency result.

3.1 First Mathematical Model Based on RX 6600 XT Graphic Card

We will start building the model by understanding basic physics. From Ohm's Law, we know that:

$$P = VI \tag{1}$$

where P denotes the total power of the electrical circuit cost, V denotes the voltage, and I denotes the electrical current. As you can see, this is a linear relationship. But this basic model does not fit well with modern processors, according to Zhang *et al.*, modern multi-core processors follow a non-linear relationship. Zhang *et al.* found that the multi-core processor under different CPU utilization could result in up to 50 percent power consumption difference, despite the samples remaining a constant operating frequency [ZLZ+15]. Since the first GPU was announced in 1999, NVIDIA's GeForce 256 standardized the design of the multi-core design of GPU with its quad-core layout [GPU22]. From the comparison between RTX 3090 Ti and RTX 3090, the result implies that we need to consider a more optimized model specialized for modern multi-core processors. There is one model derived from ohm's law that is a quadratic formula:

$$P_{dyn} = C_L V_{DD}^2 f_{0\to 1} \tag{2}$$

where P_{dyn} denotes the total power consumption by the dynamic system, C_L denotes the total capacitance on the chip, V_{DD} denotes the supply voltage, and $f_{0\to 1}$ denotes the frequency of the energy transitions from 0 to 1 of CMOS [Rea05]. This model looks more plausible in terms of non-linear relationship, but since the capacitance of the processor

changes dynamically during operation, which could only be obtained from direct calculation of the formula above to solve for C_L.

Our goal is to find a feasible mathematical model that consumers could easily approximate its power and performance, which could calculate its efficiency by applying those results to our model. For a generic graphic card, the manufacturer usually posts the following essential specifications of the product under the specification section: Compute Units (CU) or CUDA Cores (depending on the vendor), Boost/Game frequency, Video memory (VRAM) size/type, and Thermal Design Power (TDP). However, the TDP of the product does not always indicate the true maximum power — sometimes the graphic card could never achieve such high power consumption. So TDP could only count as a theoretical limit of the graphic card. The Compute unit count generally influences the performance and power consumption the most within the same generation of graphic cards since manufacturers usually use a different type of GPU die to differentiate graphic cards of different levels. AMD, for example, uses four distinct Navi 2X GPU dies: Navi 21 (max-performance), 22 (high-end), 23(mid-range), and 24 (entry-level) [GPU22]. The difference between each Navi 2X die is the different Compute Unit counts. The VRAM chip also affects power consumption since VRAM is also a type of electrical circuit. But the situation of VRAM is much more complicated: First, there are several manufacturers produce the VRAM module, which performs varies due to different manufacturing technologies; In addition, not all the graphic cards report the power draw from VRAM, which means not all the users could record the VRAM power usage from either the default graphic control panel or third party application such as AIDA 64 or MSI Afterburner.

Therefore, we must introduce a mathematical model at least involves the following variables: Voltages (V), frequency (f), Core Count (C_{total}), which is:

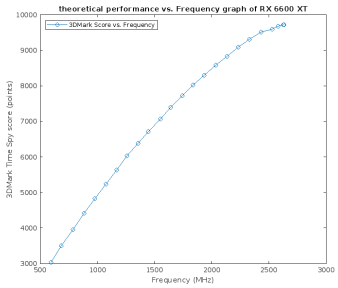
$$P_{total} = N \cdot 32 \cdot 2V^2 f + C_1 \tag{3}$$

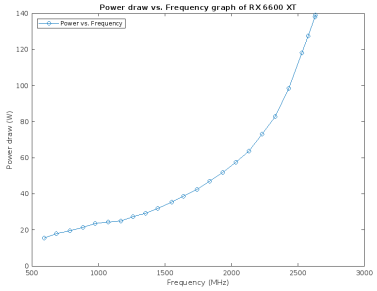
where P_{total} is the total dynamic power draw of the graphics card, N denotes the graphic card adjustment coefficient, f denotes the current core frequency, and C_1 denotes the offset constant of the graph. In unit wise, the voltage V sticks with millivolts, and frequency f is in GHz. Because we did not consider the factor involves in VRAM and other logic circuit components, the constant C_1 just be a power consumption estimate for the other components.

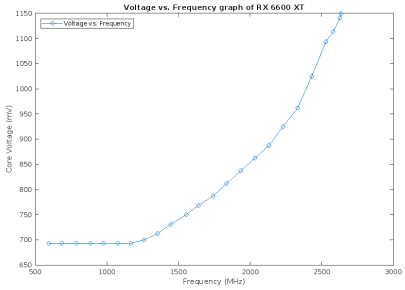
Before we dive into solving the model, we need to check whether our understanding of the relationships between the essential variables is correct or not. We set up a test platform to find the following relationships: Power vs. Fre-

quency, Performance vs. Frequency, and Voltage vs. Frequency. The test sample is the MSI Gaming AMD Radeon RX 6600 XT MECH 2X 8G OC. We use Windows 11 Pro as our test operating system. The performance test program we use is 3DMark by UL solution, which provides a fair, open platform to do benchmarks and measure hardware performance for both desktop and mobile platforms [3DM]. We use Time Spy to test its DirectX 12 performance under 2560×1440 resolution. We run the test at each designated frequency and record its power draw, voltage, and performance score. We set 5 trials for each sample frequency and took the average score. Then we generate graphs of each relationship by Matlab. The data and plots are as shown below:

Frequency (MHz)	Voltage (mV)	Static full load power(W)	Current (Calculated for reference) (A)	3DMark Graphics score	performance coefficient	perf coeff change rate	Graphic card adjustment coeff (N)	N rate
594	693	15.47	22.32323232	3034	5107.744108		0.599217425	
684	693	17.85	25.75757576	3503	5121.345029	13.6009215	0.746787616	0.147570191
786	693	19.5	28.13852814	3951	5026.717557	-94.62747199	0.786474438	0.039686822
885	693	21.52	31.05339105	4409	4981.920904	-44.7966533	0.847018236	0.060543798
977	693	23.57	34.01154401	4821	4934.493347	-47.42755697	0.903792715	0.056774479
1075	693	24.41	35.22366522	5235	4869.767442	-64.72590512	0.872246166	-0.031546549
1169	693	25	36.07503608	5635	4820.359281	-49.40816042	0.834949676	-0.03729649
1260	700	27.35	39.07142857	6025	4781.746032	-38.61324969	0.878178652	0.043228977
1355	712	29.16	40.95505618	6389	4715.129151	-66.61688045	0.871658248	-0.006520405
1447	731	31.83	43.54309166	6714	4639.944713	-75.18443809	0.882267888	0.01060964
1553	750	35.39	47.18666667	7064	4548.615583	-91.32913046	0.908277885	0.026009997
1641	768	38.65	50.32552083	7395	4506.398537	-42.21704527	0.925004034	0.016726149
1742	787	42.28	53.72299873	7722	4432.835821	-73.56271658	0.934944718	0.009940684
1837	812	47.05	57.94334975	8018	4364.725095	-68.11072563	0.955910032	0.020965314
1935	837	51.8	61.88769415	8292	4285.271318	-79.45377743	0.963594734	0.007684703
2035	862	57.6	66.82134571	8579	4215.724816	-69.5465021	0.983734787	0.020140052
2133	887	63.78	71.90529876	8834	4141.584623	-74.14019313	1.001457964	0.017723177
2231	925	72.98	78.8972973	9094	4076.199014	-65.3856087	1.031026059	0.029568095
2331	962	82.81	86.08108108	9313	3995.280995	-80.91801861	1.054747817	0.023721758
2431	1025	98.4	96	9516	3914.438503	-80.84249261	1.081607268	0.026859452
2530	1093	118.1	108.0512351	9602	3795.256917	-119.1815857	1.117673285	0.036066017
2580	1114	127.58	114.524237	9672	3748.837209	-46.41970769	1.147607532	0.029934246
2628	1141	137.97	120.9202454	9723	3699.771689	-49.0655198	1.16885743	0.021249898
2634	1150	139.15	121	9721	3690.584662	-9.187027387	1.158598551	-0.01025888
				AVERAGE:	4433.945514	-69.79411117	0.943984715	0.019520298



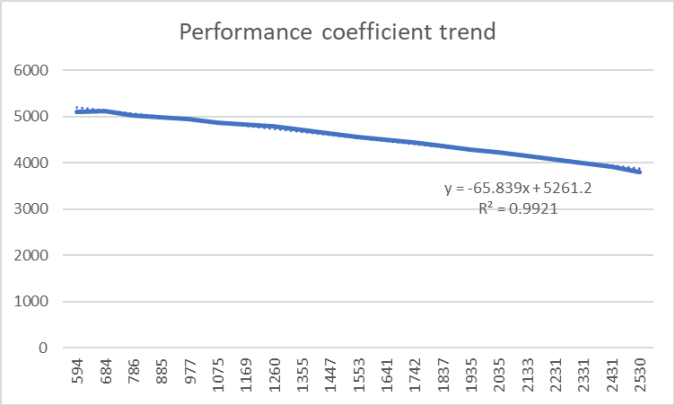




Note that we introduced the performance coefficient and its change rate as the frequency increased. The performance coefficient is calculated as (3D Mark score) / frequency (GHz). As we can see, the performance graph shows that the theoretical performance trend is linear until other components become the bottleneck, while the power graph shows that the power consumption of the graphic card is a non-linear relationship with frequency. In addition, we observe that there is a flat trend for the voltage graph at the low-frequency operation window, which indicates the initial voltage, $V_{init} = 697$ mV in this situation, which could take in consideration when we start solving the initial value problem later. In general, this card could reach 596 MHz as the lowest operating frequency and 2634 MHz as its upper bound, with a maximum voltage of 1150 mV. This card maxed out at 140 Watts due to the BIOS power limit. We also noticed that the GPU costs a small amount of power instead of zero watts at an idle state (at the lowest frequency). Recall the

power consumption model introduced before that there is a constant value C_1, which indicates the power consumption for other devices. We approximate the power draw from auxiliary devices on the GPU by simply maximum the VRAM frequency while setting the Core to idle. We recorded an average of 10 W of power draw, which we would use this value for calculation later.

After checking the relationship, we could introduce our performance model. Since we states before that this model does not take the consideration of other components other than the GPU Core, we will see that the performance will be affected by the bottleneck of other components such as VRAM frequency and bandwidth. This would cause a slower increase rate in performance when the graphic card hits a higher frequency, as shown in the Performance vs Frequency graph above. By using the data collected, we could use Excel to find its linear fit:



In the performance coefficient trend graph, the variable y denotes the performance coefficient value, and the variable $x = (f - f_{init}) \times 10$. Here we could see that the trend is very linear with $R^2 = 0.9921$. Thus, we need to introduce a limiting factor L which is defined as:

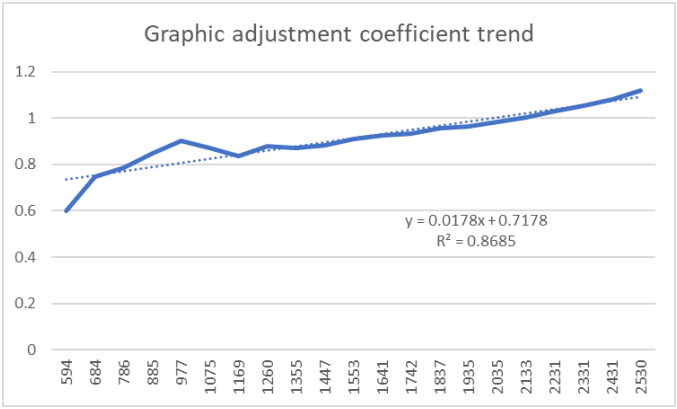
$$L = 65.839 \times 10(f - f_{init}) \tag{4}$$

where f and f_{init} represents the current operating core frequency and the initial core frequency at idle, respectively. Then we could finalized the performance model of the graphic card:

$$Perf = (5107.744 - L) \times f \tag{5}$$

From formula (3) and with the value we collected for this specific model, we could calculate an average graphic card adjustment coefficient (N). Because its voltage is constantly changing, the N value also varies when frequency changes. Similar to what we did before, we use Excel to calculate its value and its trend for each sample frequency.

The result is as shown below:



Then we define N as:

$$N = 0.0178 \times 10(f - f_{init}) + 0.7178 \tag{6}$$

Recall our goal is to find the optimum operating frequency of the GPU that gives reasonable performance while giving lower power consumption. We could introduce the criterion for this situation: The situation is considered optimal if the increase of power is not greater than the increase in performance. In mathematical language, the situation is optimal if

$$\frac{dP_{total}}{df} > \frac{dPerf}{df}$$

Then we calculate each derivative for each function defined above:

$$\frac{dP_{total}}{df} = 64V^2(0.356f - 0.178f_{init} + 0.7178)$$

$$\frac{dPerf}{df} = PC_{init} - 1316.78f + 658.39f_{init}$$

Now we could plug in the collected data into our model:

$$64V^2(0.356f - 0.178f_{init} + 0.7178) > PC_{init} - 1316.78f + 658.39f_{init}$$

$$64V^2(0.356f - 0.178(0.594) + 0.7178) > 5107.744 - 1316.78f + 658.39(0.594)$$

$$(11.392V^2 + 2633.56)f > 5498.828 - 19.586V^2$$

$$f > \frac{5498.828 - 19.586V^2}{22.784V^2 + 2633.56}$$

From the voltage data we collected, the range of voltage for this graphic card is from 0.697 V to 1.150 V. Then apply

both boundaries to solve inequalities:

$$f > \begin{cases} \frac{5498.828-19.586(0.697)^2}{22.784(0.697)^2+2633.56} & \text{when } V = 0.697 \\ \frac{5498.828-19.586(1.150)^2}{22.784(1.150)^2+2633.56} & \text{when } V = 1.150 \end{cases}$$

$$\implies f > \begin{cases} 2.076\ GHz & \text{when } V = 0.697 \\ 2.055\ GHz & \text{when } V = 1.150 \end{cases}$$

By such calculation shown above, we could reasonably deduce that any frequency inside the range $(\text{Min}(2.055, 2.076), f_{max}]$ would not consider in the optimal efficiency range. Here f_{max} denotes the maximum frequency the GPU Core could achieve in the default setting.

In other words, the optimal operation range for good power efficiency is when $f_{init} \leq f \leq \text{Min}(2.055, 2.076)$, which is reasonable that the optimal frequency range is mainly inside the lower three-fourth of the whole frequency range span. The maximum optimal frequency here is 2.055GHz, which has about 78 percent of the performance available but only consumes approximately 42.9 percent of the power compared to the situation with maximum frequency.

3.2 Generalized Model for Graphic Cards with RDNA 2 Architecture

After checking the model is feasible, we could generalize the model to fit all graphic cards variants with RDNA 2 architecture. From the AMD's official website, we know that all RX 6000 series graphic cards run with GDDR6 VRAM chips with some variation in memory capacity. So for this generalized model we will assume that the VRAM chips consumes a fixed amount of power for simplification. From the specification listed on the website, the main difference for each graphic card variant is the maximum frequency of the GPU core and the Compute Unit (CU) Core count [AMDa]. We could introduce new variables for the generalized model:

$$P_{total} = 2N \cdot CU_{count} V^2 f + C_1 \tag{7}$$

$$N = N_c \times 10(f - f_{init}) + C_2 \tag{8}$$

$$L = C_{lc} \times 10(f - f_{init}) \tag{9}$$

$$Perf = (PC_{init,G} - L) \times f \tag{10}$$

where CU_{count} denotes the total Compute Unit count for the graphic card, N_c denotes the change rate for N, C_2 denotes the linear offset for N. C_{lc} denotes the limiting coefficient of the graphic card, f and f_{init} represents the

current operating core frequency and the initial core frequency at idle, respectively. $PC_{init,G}$ is the generalized initial performance coefficient. Other variables and constants hold the same as the previous functions.

Now we can calculate the new derivatives respect to frequency f:

$$\frac{dP_{total}}{df} = (40N_cV^2f - 20N_cV^2f_{init} + 2C_2V^2)CU_{count} \tag{11}$$

$$\frac{dPerf}{df} = PC_{init,G} - 20C_{lc}f + 10C_{lc}f_{init} \tag{12}$$

We could set up the inequality just like before:

$$\begin{aligned} 40N_cV^2f - 20N_cV^2f_{init} + 2C_2V^2 &> PC_{init,G} - 20C_{lc}f + 10C_{lc}f_{init} \\ (40N_cV^2 + 20C_{lc})f &> PC_{init,G} + 20N_cV^2f_{init} - 2C_2V^2 \\ f &> \frac{PC_{init,G} + 20N_cV^2f_{init} - 2C_2V^2}{40N_cV^2 + 20C_{lc}} \end{aligned}$$

When user using this model to approximate the optimal range, user could collect Voltage (V) and minimum frequency (f_{init}) via AMD's official Control Panel. Other variables could be calculated via the method we discussed in the previous section.

For simplicity, we assume the performance of each graphic card in RDNA2 family is directly related to the CU count and frequency. We will use the data RX 6600 XT as benchmark standard. Recall that the $PC_{init} = Perf_{init}/f_{init}$, where all the variable represent the initial value for each category at minimum frequency situation. Since we assume the performance and CU count are positively related, it is reasonable to deduce that

$$PC_{init,G} = \frac{CU}{32}\frac{Perf_{init}}{f_{init}}$$

where $PC_{init,G}$ denotes the generalized initial performance coefficient, $\frac{CU}{32}$ denotes the ratio between the CU count of current sample GPU and the CU count of RX 6600 XT, which has 32 CU. We also assume each graphic card has the same minimum operating frequency as the RX 6600 XT since they use the same architecture.

Let's take a look at the RDNA 2 line-up with the specification shown below:

MODEL	COMPUTE UNITS	RAY ACCELERATORS	GAME FREQUENCY	INFINITY CACHE	MAX MEMORY SIZE	MEMORY TYPE
AMD Radeon™ RX 6950 XT	80	80	2100 MHz	128 MB	16 GB	GDDR6
AMD Radeon™ RX 6900 XT	80	80	2015 MHz	128 MB	16 GB	GDDR6
AMD Radeon™ RX 6800 XT	72	72	2015 MHz	128 MB	16 GB	GDDR6
AMD Radeon™ RX 6800	60	60	1815 MHz	128 MB	16 GB	GDDR6
AMD Radeon™ RX 6750 XT	40	40	2495 MHz	96 MB	12 GB	GDDR6
AMD Radeon™ RX 6700 XT	40	40	2424 MHz	96 MB	12 GB	GDDR6
AMD Radeon™ RX 6650 XT	32	32	2410 MHz	32 MB	8 GB	GDDR6
AMD Radeon™ RX 6700	36	36	2174 MHz	80 MB	10 GB	GDDR6
AMD Radeon™ RX 6600 XT	32	32	2359 MHz	32 MB	8 GB	GDDR6
AMD Radeon™ RX 6600	28	28	2044 MHz	32 MB	8 GB	GDDR6
AMD Radeon™ RX 6500 XT	16	16	2650 MHz	16 MB	8 GB	GDDR6
AMD Radeon™ RX 6400	12	12	2039 MHz	16 MB	4 GB	GDDR6

By plugging in CU count and frequency value, and, again, for simplicity, we will just other values would be the same as the one we measured/calculated in RX 6600 XT.

But notice that the denominator of the right hand side of the final inequality is constant here, since we assume all variables stay the same with the result from RX 6600 XT. Here we have two basic thoughts to solve this issue:

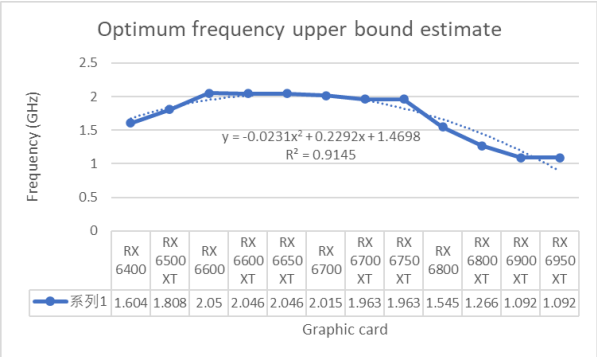
- One method is let the user define the other variable by themselves, which could potentially result more accurate output. However, the amount of work of collecting data is costly in both time and mind. It is a feasible approach for people who would devote their time into this.
- Another method is to introduce exponential parameters to the denominator to approximate the final output. It is a doubtful approach, since so far we cannot fully determine whether this approach have strong reasoning behind this adjustment. But the final result from this method is fairly close to what we have got in the previous section.

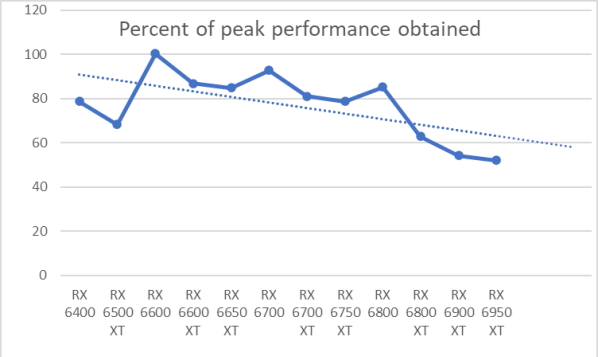
We will mainly focus on the second method beyond this point. We introduced the exponential term $\frac{3}{4}e^{CU/32}$. We choose $\frac{3}{4}$ simply because we resulted an approximately 3/4 performance when the GPU running at the upper bound of optimal range. So the adjusted formula would be like:

$$\frac{3}{4}e^{CU/32}(40CUV^2N_c + 20CU)f > (CU)(PC_{init} + 658.39f_{init} + 20CUN_cf_{init} - 2CU^2)$$

$$f > \frac{(CU)(PC_{init} + 658.39f_{init} + 20CUN_cf_{init} - 2CU^2)}{\frac{3}{4}e^{CU/32}(40CUV^2N_c + 20CU)}$$

By using Excel, we could calculate the estimated upper bound of the optimal frequency range:





From the graphs above, we could see that the average of the upper bound estimate is around 2 GHz and getting smaller as the CU count increases. It is reasonable because as the core counts increases, not only the more power it consumes but more heat it generates. The processor with more core counts has to underclock to prevent power surge and overheating due to limitation of manufacture technology. Notice that the percentage of peak performance result for RX 6600 XT in the graph above is higher than the result we obtained in the previous section. This is because the value we use for this section is the generic, official data from AMD, while the graphic card we use in the previous section is from third party vendor, which includes the custom overclock profile. The Core is the same as the original. We could observe that the performance under optimal situation keeps about 55 to 85 percent of peak performance in average, which is reasonable compare to real-world usage.

4 Limitations

There are several factors that we need to be concerned about. Firstly, we only considered the GPU Core and ignored other important components such as VRAM, IC circuit controller, VRM, thermal management components, etc. These components does not provide a interface that let us collect essential data for the model. It would be a complex task to do if we try to analyze the auxiliary components as well, since this requires some advanced knowledge in electrical engineering, which both of the team members do not. So we decided to focus on the GPU Core alone and that would come up with reasonable outcomes for us to make a mathematical model.

In addition, our math model is a simple model in term of data collection, and all the data are comes from the test named “Time Spy”. We use the application to test the GPU, and those data also not extreme accurate because there are so many specific details that could influence the results like the energy dissipation on the circuit, the software

operations in the system or even the temperature around the GPU might also interrupt the outcomes. Although we have taken each trial 5 times and then take the average, there is still small fluctuation in other areas that could potentially affect our result. It could be biased with the application itself because the application aims to test the certain scenario, the performance score could be unjust. The application maybe good on test on specific area of a GPU, but not including all areas of a GPU. Therefore, the result may be affected if a GPU meets a scenario that does not take its advantage, while the score may be unrealistically high if the GPU is specifically optimized in such area.

Moreover, those graphs and data we collected could only represent the RDNA 2 architecture but not the other architecture like Turing and Ampere, so that means our math model only could predict the performance and frequency on RDNA 2 architecture not that of the other architectures. Other architectures have different manufacturing technology and different structures which could make the generalization becomes complex and very difficult.

Finally, the model itself may be flawed. We developed our model based on a specific variant of RDNA 2 family. When we tried to generalize it to fit all variants of RDNA 2 graphic cards, we did a several assumptions which should not be fixed in such situation. Each GPU variant has its own characteristics in power control and performance management. We generalized based on the RX 6600 XT cannot perfectly fit with other graphic cards in the same family even they share the same architecture properties.

5 Conclusion

This research is motivated by the recent developments in the GPU industry. We used the math model to predict the performance of GPU built with RDNA 2 architecture. After we ran the test, we found out that the power consumption started to raise at a higher rate than its performance increase rate after passing a specific limit. We introduced two custom formulas with custom variables and parameters to approximate the power usage and performance of the graphic card at a given frequency. Then we use the corresponding derivative and set up an inequality to find the turning point when the power increase rate is greater than its performance increase trend. From the result we calculated and compared with the official maximum frequency for each sample graphic card, it seems normal that GPU vendors tend to maximize performance for each GPU with a sacrifice in an increase of power consumption, usually 2 times higher than the power consumption in optimal condition we calculated. This result reflects two aspects: First, it becomes a convention that each GPU vendor tends to maximize performance to be more competitive in the consumer market

while lack of balancing power efficiency; Second, we see the optimal range of GPU with more core counts is narrower than the one with fewer core counts, which not only indicates the physics law we need to consider but also considers us to either further improve the existing architecture by manufacturing with more advanced technology or develop a new architecture which could improve efficiency in some scale in the future.

Regarding limitations, as we discussed before, our mathematical model only represents samples from the consumer market, which means this may not be suitable for the industrial market with Pro-series GPUs. The professional graphic card usually has a much better balance between performance and power usage to benefit industrial profit, which is quite different compared to the consumer market.

As our world keeps fast pace in advancing modern technologies, new manufacturing technologies may affect power efficiency and form more powerful GPUs while maintaining reasonable power efficiency. This means our model may fail when dealing with newer samples in the future. In future research, we can focus more on the characteristic of exceeding power consumption. If the power consumption and the frequency are still in an exponential relationship, the power needed would be unacceptable high, and the related components such as the cooling system also have to be upgraded and make their price too high to use in consumer's perspective. As the wheels of the technology keep rolling forward, we need to keep close attention to the finer technology which may bring us a well-balanced product, which we may shift our focus from efficiency to other perspectives. Under the current situation, however, we will still focus on improving efficiency. As we learn more knowledge in this area, we may compare the differences of various architectures at the hardware level, which could help us to develop a well-defined accurate mathematical model with advanced knowledge background.

References

[3DM] 3dmark.com - share and compare scores from ul solutions. https://www.3dmark.com/. Last accessed on 2022/8/12.

[Als22] Thomas Alsop. Share of households with a computer worldwide 2005-2019. 2022. https://www.statista.com/statistics/748551/worldwide-households-with-computer/. Last accessed on 2022-8-6.

[AMDa] AMD. Amd radeon™ rx graphics cards. https://www.amd.com/en/graphics/radeon-rx-graphics, Last accessed on 2022/8/15.

[AMDb] AMD. Amd rdna™ 2 graphics architecture. https://www.amd.com/en/technologies/rdna-2. Last accessed on 2022/8/6.

[Bur47] A.W. Burks. Electronic computing circuits of the eniac. *Proceedings of the IRE*, 35(8):756–767, 1947.

[GPU22] Gpu specs database. July 2022. https://www.techpowerup.com/gpu-specs/. Last accessed on 2022/8/12.

[Int] Intel. Intel® iris® xe and iris® plus graphics. https://www.intel.com/content/www/us/en/architecture-and-technology/visual-technology/graphics-overview.html. Last accessed on 2022-8-6.

[KB14] Jon H. Kaas and Pooja Balaram. Current research on the organization and function of the visual system in primates. *Eye Brain*, 6:1–4, 2014.

[Mic] Microsoft. All about graphics processing units (gpus). https://support.microsoft.com/en-us/windows/all-about-graphics-processing-units-gpus-e159bedb-80b7-4738-a0c1-76d2a05b eab4. Last accessed on 2022-8-6.

[Nvi] Nvidia. Nvidia ampere architecture: The heart of the modern data center. https://www.nvidia.com/en-us/data-center/ampere-architecture/. Last accessed on 2022-8-6.

[Rea05] Jan M. Rabaey et al. *Digital Integrated Circuits: A Design Perspective*. Pearson Education, Upper Saddle River, NJ, 2nd edition ed. edition, 2005.

[Wal] Steven Walton. Nvidia geforce rtx 3090 ti review. https://www.techspot.com/review/2442-geforce-rtx-3090-ti/. Last accessed on 2022/8/10.

[ZLZ+15] Yifan Zhang, Yunxin Liu, Li Zhuang, Xuanzhe Liu, Feng Zhao, and Qun Li. Accurate cpu power modeling for multicore smartphones. Technical Report MSR-TR-2015-9, February 2015.