2014

What We Are Paying for: A Quality Adjusted Price Index for Laptop Microprocessors

Liyang Sun
lsun@wellesley.edu

What We Are Paying for

A Quality Adjusted Price Index for Laptop Microprocessors

Sophie Sun

Submitted in Partial Fulfillment of the Prerequisite for Honors in Economics

April 2014

What We Are Paying for

A Quality Adjusted Price Index for Laptop Microprocessors

Senior Thesis Draft

Author:
Sophie Sun '14
Wellesley College

Supervisor:
Professor Daniel E. Sichel
Wellesley College

April 25, 2014

Abstract

A microprocessor contains the central processing unit and takes the role of the "brain" for a computer. For the past decades, we have benefited greatly from its technological improvement. To accurately measure the contribution of such technological improvement to economic growth, we need a quality adjusted price index, which also helps us understand quality and technology trends in microprocessors. The quality trend in desktop microprocessors has been extensively studied. I focus on microprocessors for laptops for my senior economics thesis. Using data I newly collected on laptop microprocessor prices and performance metrics, I construct a quality adjusted price index spanning the past ten years. Across a range of empirical specifications, I find a sharp decrease in quality adjusted price over 2004-2013, but smaller in magnitude since 2010. These results might suggest a different technological improvement pattern and/or changing pricing strategies in the laptop microprocessor segment of the industry.

Acknowledgments

I am highly indebted to Prof. Sichel for his guidance and constant support throughout this senior thesis. It is Prof. Sichel who inspired me with this thesis topic and provides me with the possibility to complete this study.

Special thanks goes to Dr. David Byrne and Dr. Steve Oliner, who provided enormous help with data collection.

I would also like to thank Lizi Chen '13, Cirrus Foroughi, Yikang Li '14, Christine Park '14, Prof. Casey Pattanayak, John Shi, Cory Smith, Wendy Wu '14, whose contribution in suggestions and encouragement helped me to complete the writing. I thank Helen Willis '14 for sheltering me at her thesis carrel.

I offer my sincere appreciation for the learning opportunities provided by the Economic Research Seminar directed by Prof. Eric Hilt.

Last but not least, I would like to express my gratitude toward my family.

All errors are solely mine.

Contents

1 Introduction

The world has benefited greatly from the Information Revolution, from the ease of online shopping to the accuracy of medical imaging. Microprocessors, the small semiconductor chips in our computers, tablets, smartphones and other electronic devices, play a central role in these developments. Microprocessors function as a computer's brain, processing the tasks assigned by the users. The first microprocessor was introduced by Intel commercially in 1971: Intel 4004 at a price of $60.[1] Since then, innovations in production processes have driven down the cost of manufacturing, such that microprocessors became a lot cheaper and therefore have made Information Technology (IT) more accessible in everyday life. According to the Census Bureau, 75.6 percent of households in America reported having a computer in 2011, compared with only 8.2 percent in 1984.[2] More than that, microprocessors are drastically better today than forty years ago in terms of quality. The conventional evaluation of a microprocessor's quality compares the number of transistors, the tiny electrical switches made of silicon on the microprocessor. As a result of constantly shrinking the transistor sizes, technology nowadays can easily enable billions of transistors to fit on one microprocessor, compared with only thousands in early days of microprocessors. Such improvement has increased the computing power of microprocessors substantially. The design of microprocessors has become more sophisticated at the same time. Betker et al. [1997] remind us that the first microprocessor was intended for an electronic calculator. Nowadays microprocessors are general-purposed, capable of running a wide range of application software.

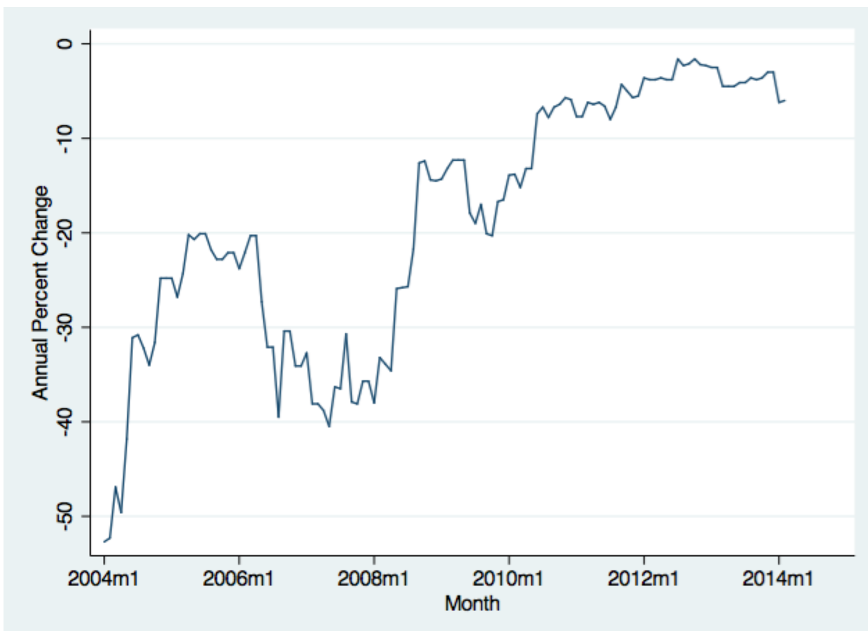
Besides its contribution to the standard of living, progress in the microprocessor industry is a key driver of the phenomenal productivity growth over the past few decades. To accurately measure real input and output in the microprocessor industry in national accounts, we need quality adjusted price indices, which account for price changes while controlling for quality. Two popular methods for computing quality adjusted price indices are matched-model and hedonic methods. The matched-model method traces the prices for one model of goods with constant quality features over time and averages price relatives for all of the models traced in a period. As is common in statistical agencies, official price indices such as the Producer Price Index (PPI) by the Bureau of

[1] This price converts to about $350 in 2013 dollars.

[2] See *Computer and Internet Use in the United States* (2011 Report), available at `http://www.census.gov/prod/2013pubs/p20-569.pdf`.

Labor Statistics (BLS) for many products are derived based on the matched-model method. The hedonic method, on the other hand, directly controls for quality changes. Previous work by Byrne et al. [2014] has shown that the matched-model method used by BLS for the microprocessor PPI might be biased. Rather than the slow decline post-2010 depicted in Figure 1, a hedonic index of desktop microprocessors experiences a much sharper decline. This discrepancy calls into question the validity of the microprocessor PPI. However, the verdict on PPI also depends on a less studied question: for laptop microprocessors, is the PPI also biased? The contribution of this paper is to provide empirical evidence on this question by estimating quality adjusted price indices for laptop microprocessors.

Figure 1: BLS Microprocessor PPI



Notes: Source: Bureau of Labor Statistics Producer Price Index Industry Data.[3]

Using public online information, I construct a new dataset of microprocessor prices and quality metrics. These data allow me to construct quality adjusted price indices over 2004-2013. Across a range of empirical specifications, I find a sharp decrease in price over 2004-2013 but smaller in magnitude since 2010. These results suggest a different technological improvement pattern. Also, there is qualitative evidence on potential changes in pricing strategies in the laptop microprocessor

[3]BLS PPI extracted on: April 9, 2014. Industry: Semiconductors and related device mfg; Product: Microprocessors (including microcontrollers); Series Id: PCU33441333441312; Base Date: 200706. The BLS sample includes microprocessors in servers, desktops, laptops, and other microprocessors.

segment of the industry.

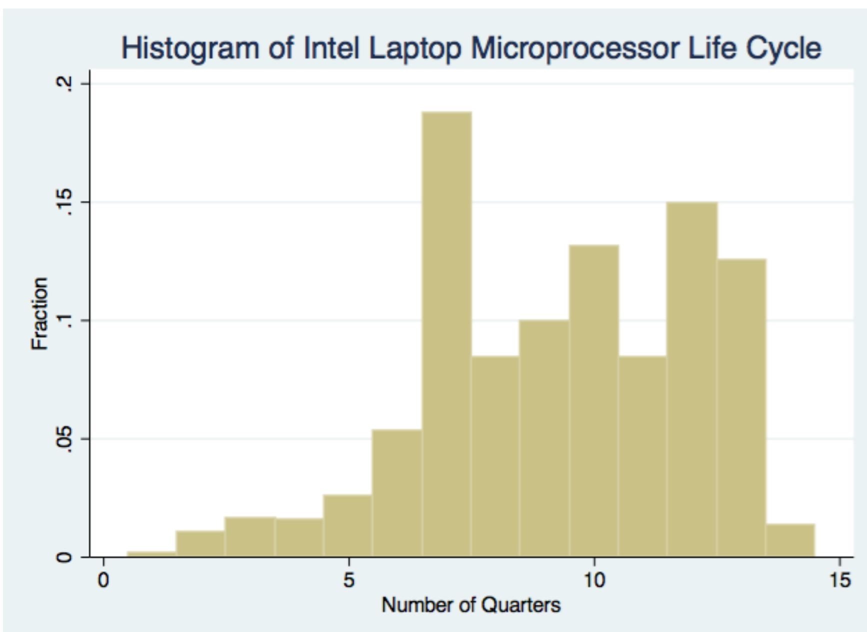
The rest of the paper proceeds as follows. Section 2 introduces the empirical approaches and describes the range of previous literature concerning quality adjusted price indices for microprocessors. Section 3 describes the microprocessor market and the data. Section 4 presents estimates of hedonic indices for laptop microprocessors and attempts to track the quality trend in the microprocessor industry. Section 5 examines the robustness of the empirical results. Section 6 concludes.

2 Preliminaries on Price Indices

2.1 New Goods Bias in the Matched-Model Method

The discussion of quality adjusted price indices centers around the difference between matched-model and hedonic methods. The matched-model index is constructed by averaging the prices changes of the goods in a sample whose quality features stay the same over adjacent time periods. This is a common practice by statistical agencies. The theory behind the matched-model is intuitive and the required data collection is straightforward. However, the matched-model index also introduces "new goods bias." Since there are always goods exiting or entering the market, the sample of constant quality goods never remains the same over time. This bias is especially notable in the microprocessors market, where life cycle ranges from 5 to 13 quarters as shown in Figure 2.

Figure 2: Histogram of Intel Laptop Microprocessor Lifecycle



Notes: The 95% interval is [5,13]. Year = 2004-2013.

The problem as discussed in Pakes [2003] is that the matched-model method fails to make any adjustment for differences between the "utility per dollar" of the new and old goods. For example, when Intel introduced multicore microprocessors, the single-core microprocessors became obsolete. However, consumers would gradually switch from the old single-core to the new multicore microprocessors, rather than shifting completely at the introduction of multicore microprocessors. Any consumer who switches experiences a price change that is not equal to the change in prices between the new and the old microprocessors, and consumers increase their utility as a result of the switch. This utility gain is not captured by matched-model indices so it results in a new goods bias, which tends to introduce an upward bias into the estimates of changes in the matched-model indices when the new goods are better in terms of quality.

2.2 Overview of Hedonic Method

The alternative method, the hedonic method, can ameliorate the new goods bias. The hedonic method utilizes hedonic functions, which provide an explicit way to control for quality changes. A hedonic function is a relation between the prices of different goods in a sample, and the quantities of quality features in them. For example, a hedonic function can capture ion the empirical relationship between the prices of a set of microprocessors and how many transistors there are on the microprocessors. Different forms of hedonic functions allow for different price index formulae. Therefore there are a number of ways to compute hedonic indices depending on the forms of hedonic functions. I now provide a brief overview of hedonic functions. Following the notation in Pakes [2003] , I denote (x_i, p_i) to be the quality features and the price of good i and (x_{-i}, p_{-i}) to be the quality features and prices of the other goods marketed. Then the demand for good i becomes $D_i(\cdot) = D(x_i, p_i, x_{-i}, p_{-i}, A)$ where A is the distribution of consumer attributes that determine consumers' preferences over quality features. Suppose all firms are single good firms and marginal costs are given by $mc(\cdot)$, then prices are $p_i = mc(\cdot) + \frac{D_i(\cdot)}{|\partial D_i(\cdot)/\partial p|}$. The second term $\frac{D_i(\cdot)}{|\partial D_i(\cdot)/\partial p|}$ is the mark-up which varies inversely with the elasticity of demand. The hedonic function $h(x)$ is the expectation of price conditioned on quality features x_i, that is $E[p_i|x_i] = E(mc(\cdot)|x_i) + E(\frac{D_i(\cdot)}{|\partial D_i(\cdot)/\partial p|}|x_i)$. So the hedonic function is the expectation of marginal costs plus that of the mark-up conditioned on the good's quality features. The underlying assumption for the hedonic function is that the quality features are costly to produce, and consumers make purchasing choices based on the quality features. Let $h^t(x_i)$ be the hedonic function in period t.

2.3 Alternative Approaches to Hedonic Price Indices

With these preliminaries, I now introduce hedonic price indices. As defined earlier, a hedonic price index is a price index that uses a hedonic function in some way.

2.3.1 Dummy-Variable Method

The first is the dummy variable (DV) method. Pooling prices and quality features of goods in the sample over time, this method controls for quality features, and assigns dummy variables for each time period in which a price is observed for the good. Hence, there is only one hedonic function $h_{DV}(x_i)$ for the sample. A stylized dummy variable hedonic function takes the form

$$h_{DV}(x_i) : ln(P_{i,t}) = \alpha + \beta ln(x_i) + \sum_t \delta_t D_{i,t} + \epsilon_{i,t}, \tag{1}$$

where on the left-hand side $P_{i,t}$ is the observed price of good i at time period t, on the right-hand side, the x_i is the time-invariant quality features for good i. The β coefficient measure by how much the price changes corresponding to 1% change in x_i. It is constrained to be the same over all time periods, which is in effect an average of the quality elasticity of price for each of the time periods in the sample. The time dummy variable $D_{i,t}$ is a constructed binary variable which equals 1 if a price is observed for good i at time period t, and 0 otherwise. I exclude $D_{i,t=1}$ to ensure the initial time period is the baseline. The DV method entails an unweighted geometric mean index number formula. To illustrate this point, I expand Equation 1 as in Triplett [2004]

$$h_{DV}(x_i) = \begin{cases} ln(P_{i,1}) & = \alpha + \beta ln(x_i) + \epsilon_{i,t} \\ ln(P_{i,2}) & = \alpha + \beta ln(x_i) + \delta_2(D_{i,2} = 1) + \epsilon_{i,t} \\ \dots & \dots \\ ln(P_{i,n}) & = \alpha + \beta ln(x_i) + \delta_n(D_{i,n} = 1) + \epsilon_{i,t} \\ \dots & \dots \end{cases}$$

The δ_n is the log difference in prices between $t = n$ and $t = 1$, holding constant quality features x_i of the goods. Similarly, the difference in prices between two adjacent periods is $\delta_t - \delta_{t-1}$, holding constant quality features x_i of the goods. It represents the price change in time period t

that is not associated with changes in quality features.

Denote N_t to be the number of goods that have price information at time period t. Then I note that $\delta_t - \delta_{t-1}$ as $\frac{1}{N_t}\sum_{n \in N_t}(\ln P_{n,t} - \beta ln(x_n)) - \frac{1}{N_{t-1}}\sum_{n \in N_{t-1}}(\ln P_{n,t-1} - \beta ln(x_n))$. Then the implicit index number formula is defined as

$$index\{\frac{t}{t-1}\} = exp(\delta_t - \delta_{t-1}) = \frac{\dfrac{\prod_{n \in N_t}(P_{n,t})^{1/N_t}}{\prod_{n \in N_{t-1}}(P_{n,t-1})^{1/N_{t-1}}}}{\dfrac{\prod_{n \in N_t}(x_n)^{\beta/N_t}}{\prod_{n \in N_{t-1}}(x_n)^{\beta/N_{t-1}}}}. \tag{2}$$

Equation 2 shows the DV index equals the ratio of unweighted geometric means of prices in adjacent periods, divided by a hedonic quality adjustment. The denominator of Equation 2 is the hedonic quality adjustment, which is itself an index number: a quantity index that measures the change in quality features of goods sold in adjacent periods as explained in Triplett [2004]. When N and x_n are constant from t to $t-1$, then the DV index is the geometric mean of prices. That is, the DV index would yield same results as the matched-model geometric mean index when there are no new goods entering or old goods exiting the market, which Aizcorbe et al. [2003] confirm with empirical evidence. I include the mathematical proof in the Appendix.

If $h_{DV}(x_i)$ is estimated with weights by sales, the DV index as in Equation 2 would be a ratio of weighted geometric means, rather than the equally-weighted geometric means. However, the form of $h_{DV}(x_i)$ imposes the geometric means in the index number formula. It is not possible to calculate a superlative index number formula, independently from $h_{DV}(x_i)$. The second hedonic function introduced here, the characteristic method, relaxes this constraint.

2.3.2 Characteristics Method

The characteristics method uses the implicit characteristic prices estimated from hedonic regressions in a conventional weighted index number formula. Following the notation in Triplett [2004], for each time period t in the sample, the hedonic function $h^t_{char}(x_{i,t})$ takes the form

$$h^t_{char} : ln(P_{i,t}) \quad = \quad \alpha_t + \beta_t ln(x_{i,t}) + \epsilon_{i,t},$$

where on the left-hand side $P_{i,t}$ is the observed price of good i at time period t, on the right-hand side, the $x_{i,t}$ is the time-invariant quality features for good i marketed at time period t. "Characteristics" are the same as the quality features, just as the name "characteristics method" suggests. The β measures by how much the price changes corresponding to a 1% change in x_i at time period t in the sample.

The predicted value $\hat{h^t_{char}}(x_i, t)$ is the implicit characteristic price.[4] For two overlapping periods $t + 1$ and $t + 2$, one can obtain the implicit characteristics prices from both periods by estimating

$$\begin{aligned} h^{t+1}_{char} : ln(P_{i,t+1}) &= \alpha_{t+1} + \beta_{t+1} ln(x_{i,t+1}) + \epsilon_{i,t+1} \\ h^{t+2}_{char} : ln(P_{i,t+2}) &= \alpha_{t+2} + \beta_{t+2} ln(x_{i,t+2}) + \epsilon_{i,t+2}. \end{aligned}$$

Using the characteristic prices, one can choose the index number formula and construct indices. I briefly present three formulae that I use.

The Laspeyres index, which is the base period weighted index, takes the form

$$\frac{\hat{h^{t+2}_{char}}(x_{i,t+1})}{\hat{h^{t+1}_{char}}(x_{i,t+1})}, \tag{3}$$

where the numerator of equation 3 is constructed from the characteristics of the good in the initial

[4] When exponentiating the regression prediction, which is the predicted log of price, there is a correction $exp(\epsilon_{i,t})$ which I assume equals $exp(0.5\hat{Var}(\epsilon_{i,t}))$ as suggested by Pakes [2003]. The underlying assumption is that the error terms are from a log normal distribution.

period $t+1$ valued by the second period's hedonic function, that is, the second period $t+2$ characteristics prices for goods in the initial period. The denominator uses the initial period $t+1$ characteristics prices for goods in the initial period. In a PPI context, equation 3 could be thought of in terms of the total quantity of characteristics produced by the industry in the initial period. The numerator values the initial period's output of characteristics with characteristics prices of the second period; it is the industry's hypothetical revenue if it sold the initial period's goods at the characteristics prices that prevail in the second period. The denominator is the actual industry revenue in the initial period.

The Paasche index, which is a current-period weighted index, takes the analogous form

$$\frac{h_{char}^{t+2}(\hat{x}_{i,t+2})}{h_{char}^{t+1}(\hat{x}_{i,t+2})}. \tag{4}$$

The Fisher index, which is the chain weighted index, is the geometric mean of Laspeyres and Paasche indices:

$$\left(\frac{h_{char}^{t+2}(\hat{x}_{i,t+1})}{h_{char}^{t+1}(\hat{x}_{i,t+1})} \cdot \frac{h_{char}^{t+2}(\hat{x}_{i,t+2})}{h_{char}^{t+1}(\hat{x}_{i,t+2})}.\right)^{1/2} \tag{5}$$

To summarize, the Laspeyres index draws characteristics from the initial period to "forecast" to in the second period. Paasche draws characteristics from the second period and "backcast" prices in the initial period. The Fisher index values characteristics bundles from both periods.

In this paper, I apply the indices to the average good in each period, i.e. I take the mean of the predicted prices for all goods in one period, and use this average price as the index for the period.[5] Triplett [2004] points out that this average good might not exist; it is merely the good with the average quantity of characteristics purchased in one period. For example, the "average" microprocessor may be a microprocessor with the average number of transistors among all microprocessors marketed in one period. Such a microprocessor might not exist in the actual market, but I take this hypothetical microprocessor as the representative of the market in one period.

Similarly to the DV method, when there are no new goods entering or old goods exiting the

[5] Precisely I take the geometric mean of the indices since the original scale is in log scale, i.e. the average of logs of prices converts to the geometric mean of prices.

market, the characteristics method would yield same results as the matched model index. I include the mathematical proof in the Appendix.

2.3.3 Adjacent-Period Method

However, given the constraint on data availability, the characteristics method might not be feasible. If there are too few observations within certain time periods, the characteristics method could generate unreliable estimates. To alleviate this shortcoming, Byrne et al. [2014] suggest an alternative method, the adjacent-period method. The essence of the adjacent-period method is a DV method for each two overlapping period, which takes the form

$$h_{adj}^{\{t_j,t_{j+1}\}}(x_{i,t}) : ln(P_{i,t}) = \alpha + \beta ln(x_{i,t}) + \sum_t \delta_t D_{i,t} + \epsilon_{i,t}, \tag{6}$$

where $t \in \{t_j, t_{j+1}\}$, the two overlapping periods. The rest of the setup is identical to the DV method in Section 2.3.1. To construct the index, the adjacent-period method can either infer the growth rates of prices from the δ's, or can predict the prices and apply some index formula similarly as characteristics method. In my analysis I choose to infer price index from the δ's. Again, when there are no new goods entering or old goods exiting the market, the adjacent-period method would yield very similar results to the matched-model index.

In Section 4.2, I compute quality adjusted price indices using all three methods and discuss their strengths and weakness in an empirical context.

2.4 Previous Literature on Microprocessor Price Indices

My work extends previous research on microprocessor hedonic price indices. Grimm [1998] calculate that the price index for microprocessors declined at a 35% average annual rate from 1985 to 1996. For 1990-1998, Chwelos [2003] estimates laptop prices decline at an average rate of 40% per year. Although the prices were for laptop computers, Chwelos [2003] relies on quality features of microprocessors. Therefore his estimates are reflective of the price trend in laptop microprocessors. Other research focused on either the entire microprocessor market or desktop microprocessors, and documented similar trends for the 1990s.

Quality adjusted prices for microprocessors continued to decline rapidly in the early 2000s.

Aizcorbe et al. [2008] estimate the quality adjusted price for microprocessors declined 40.5% per year on average over 2001-2004 using the matched-model method. Recent work by Byrne et al. [2014] estimate desktop microprocessor prices declined an average of 44% over 2000-2012 using the hedonic adjacent-period method. They find that both methods, the matched model method and the hedonic method, yield similar indices for the early 2000s.

I now examine the 2004-2013 laptop microprocessor mark. I present empirical estimates of quality adjusted price indices and quality improvement in Section 4.2. Before that, I firstly describe the microprocessor market and data collection in the next section.

3 Empirical Context

This section briefly reviews the microprocessor market and the data construction, presenting information relevant for the empirical analysis. The Appendix documents more details on the data collection and technical background.

3.1 Laptop Microprocessor Market

Intel and Advanced Micro Devices (AMD) dominate the market for laptop microprocessors. The industry is highly competitive regarding technical leadership, which Intel gained in 2006 with the release of the Core 2 family. For laptop microprocessors, Intel now leads the market with more than 90% market share according to Byrne et al. [2014]. However, while Intel controls the high-end and mid-range parts of the market, AMD remains a competitor for the low-end parts of the market.

At any given time, manufacturers offer a variety of microprocessors. In particular, Intel now introduces new microprocessors more widely across its product line, improving high-end and low-end microprocessors, a new strategy for Intel in the past decade. The most important quality feature of these microprocessors is performance, the ability to process tasks. High performing microprocessors tend to have more transistors, include multiple cores, and are equipped with more advanced designs. Engineers can change all of these quality features to manipulate performance, but a simple way to compare performances is based on some benchmark scores.

3.2 Tracking Prices

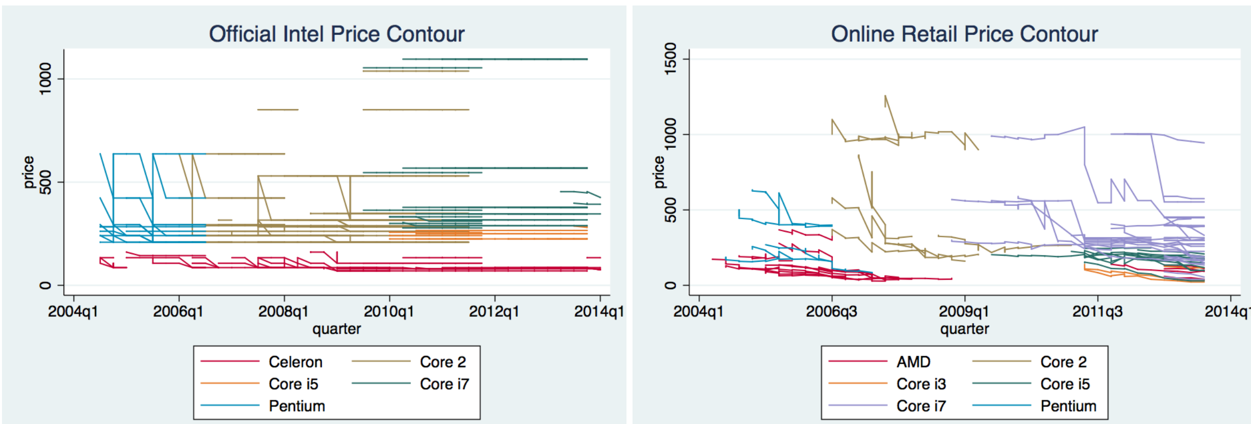
Microprocessors prices are gathered from two sources. Intel publishes bi-monthly price lists on its investor relation website. These are wholesale list prices for microprocessors sold in units of 1,000, also called "tray" prices as microprocessors are packaged on a tray. I am able to access the majority of these price lists for 2004-2013.[6] Prices from 2004-2013 were also gathered from websites devoted to tracking retail prices for Intel and AMD microprocessors. I relied mostly on SharkyExtreme and Pricewatch.[7] These price search engines monitor major online retail market places such as Amazon, eBay and Newegg for price quotes and track the lowest price. The frequency

[6] Collected by Dr. David Byrne and the author.

[7] Collected by Dr. David Byrne and the author. Since Pricewatch gets updated daily, I exploit the "wayback machine", an internet archive, for past versions of the same website. See http://web.archive.org/web/*/http://www.pricewatch.com/cpu/

of price quotes ranges from bi-monthly to quarterly as shown in Table A1 in the Appendix. The frequency I observe a price quote depends largely on data availability, not the popularity of the microprocessor. In particular, multiple price quotes in a given period might not be associated with multiple transactions and market popularity for the particular microprocessor. Therefore a consistent frequency of price quotes can correct this kind of bias in later analysis. I construct quarterly prices using the minimum price within each quarter for each microprocessor.

Figure 3: Laptop Microprocessor Price Contour



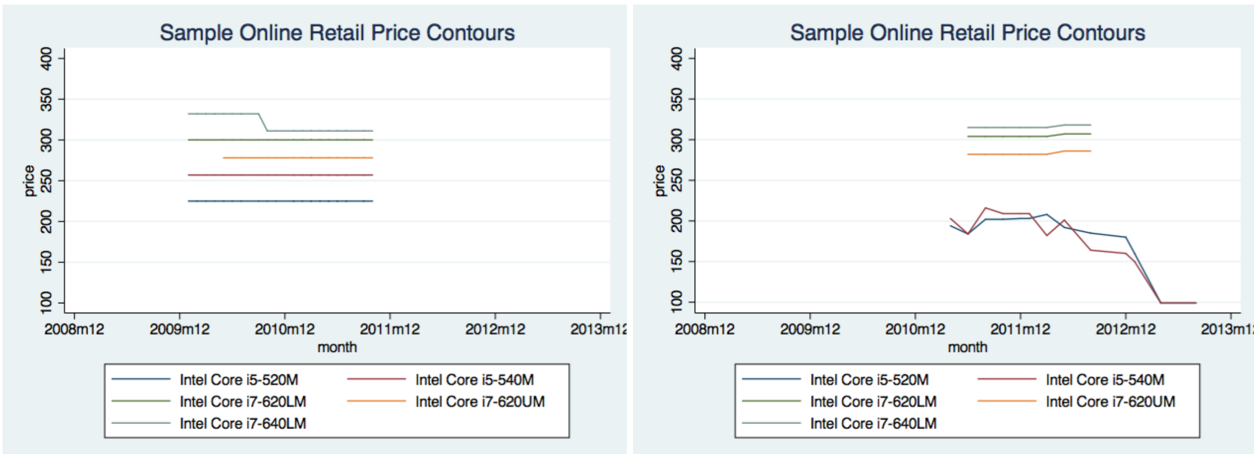
Notes: Each line traces the price for a microprocessor. Each color represent a family of microprocessors.

Figure 3 plots the price contours and Table A2 in the Appendix details the price distribution by year and by family. The two price sources represent different segments of the laptop microprocessor market. Intel's wholesale price lists provide a more comprehensive sample with 170 microprocessors. Among the sample of 88 microprocessors, Online retail websites list prices for some desktop microprocessors that are used in high-end laptops. Figure 4 compares the prices for five specific microprocessors, which highlights the differences between the two price sources. There is certainly less fluctuation in Intel prices, and in some extreme cases prices remain constant throughout the model's lifecycle, even when the retail prices are dropping sharply. This contrast casts doubt on the validity of Intel prices. Intel might have not posted actual transaction prices after the introduction of the microprocessors. The estimates on quality adjusted prices would be biased if these prices do not represent transaction prices. To ameliorate such bias, I will use the price from the first time period, that is, the introduction price of each microprocessor.[8] Therefore

[8]This pricing strategy might be due to the multi-product marketing by Intel as discussed in Nosko [2010]. Byrne

in the analysis, the Intel price dataset has one price per microprocessor, whereas the online retail price dataset has a quarterly price series per microprocessor.

Figure 4: Selected Laptop Microprocessor Price Contours



3.3 Measuring Performance

For quality features, one might compare microprocessor specifications, which are basic technology information about the microprocessor.[9] Essential specifications include number of transistors, lithography (size of transistor), clock speed (processing frequency), number of cores and architecture (microprocessor design). These specifications are inadequate quality features as they only partially represent microprocessors' performance. Some of the previous empirical research has used benchmark scores to control for quality in developing quality adjusted prices index for microprocessor and/or computers.

Benchmark software assigns the microprocessor standard tasks and scores the performance. There are multiple benchmarks available targeting different features of microprocessors. I focus on two main features: basic performance and graphics performance.[10] By basic performance, I

et al. [2014] offers a fuller justification for using introduction prices given Intel's pattern of posted prices.

[9]The information on microprocessor specifications are collected from microprocessor website YouCPU (see `http://www.youcpu.com/en/`). , and verified on Intel and AMD official websites. More technical information can be found in the Appendix.

[10]It is arguable whether graphics performance is entirely a microprocessor feature. A graphics processing unit (GPU) can be on a video card, on the motherboard along with the microprocessor, or on the microprocessor. It was only recently that Intel and AMD began to commonly integrate GPU into microprocessor architecture (2010 for Intel – Westmere architecture and 2011 for AMD – Accelerated Processing Unit (APU) Series). With a discrete GPU, to measure graphics performance, ideally I would need GPU information as well, which I postpone as a future research topic.

refer to the microprocessor's ability to process usual tasks such as running ordinary software applications. By graphics performance, I refer to the microprocessor's ability to render 3D images such as streaming videos.[11] I argue that both performances are essential quality features that an average consumer would factor in their purchase decision process. Possibly the recent popularity of video websites has induced a more central role of graphics performance, and at the same time, the perfection of basic microprocessor performance has satiated consumers' needs. I choose three benchmarks, SuperPI1M, wPrime32 and 3DMark06.[12] For basic performance, I choose SuperPI1M and wPrime32, for they are free, popular and straightforward. For graphics performance, I choose 3DMark06, for I am able to obtain a reasonable overlap on microprocessors between price information and 3DMark06 scores.[13]

SuperPI1M uses single-threading method to calculate π to a million decimal places. It tests the ability of a microprocessor to complete one task at a time. wPrime32 uses multi-threading method to estimate the square roots of first 32 million integers. It tests the ability of a microprocessor to multitask at a time. Both tasks, calculating π and square-rooting integers, are straightforward tasks for microprocessors and resemble ordinary software applications. Scores from these two benchmarks are the numbers of seconds taken to complete the assignments. Lower scores in SuperPI1M and wPrime32 indicate better basic performances for microprocessors in single-threading or multi-threading ability. 3DMark06 provides a set of tests, mainly graphics tests, and scores based on the number of frames per second rendered by the microprocessor. Higher 3DMark06 scores indicate a better graphics performance.

Since the benchmark scores provide a comprehensive comparison among microprocessors, computer hobbyists have charted the results extensively, and I obtain these data from a hobbyist website.[14] Table A3 in the Appendix shows the coverage of these benchmark scores after matching with prices by microprocessor name. Not all laptop microprocessors are benchmarked. Therefore not all laptop microprocessors can be matched with a benchmark score. Note that the sample size

[11] Graphics performance may also reflect the ability to process large tasks. GPUs can be more effective than CPUs at processing parallel algorithms because of their highly parallelized structures.

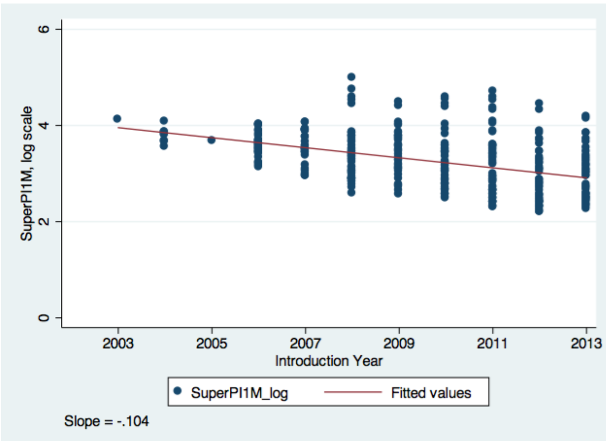
[12] These three benchmarks are only available for Windows. Since operating system affects microprocessor performance, the limitations of these two benchmark in fact remove the bias that is due to different operating systems.

[13] Nonetheless, the newer version of 3DMark06, 3DMark11, and some other benchmarks such as Cinebench, are more accurate at capturing graphics performance. See the Appendix for details on benchmarking.

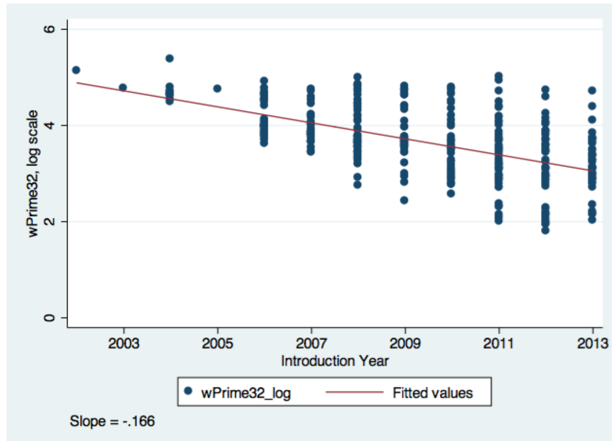
[14] NotebookCheck, which collects the average of scores from benchmark websites (SuperPI at `http://www.superpi.net/`, wPrime at `http://www.wprime.net/` and 3DMark at `http://www.futuremark.com/benchmarks/3dmark`). See the Appendix for details on benchmarking.

of matched microprocessors varies by benchmark as well. Some microprocessors with SuperPI1M scores are missing wPrime32 scores, etc.

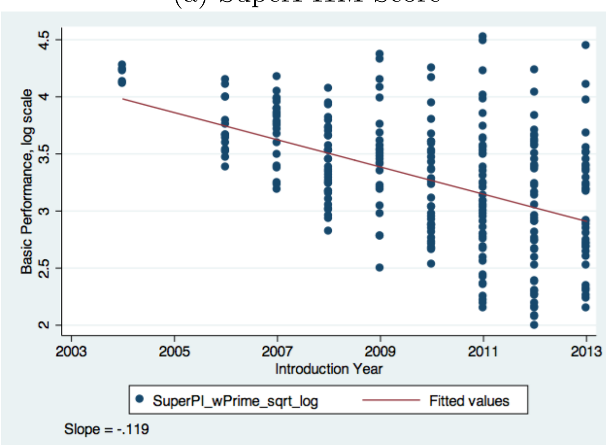
Figure 5: Laptop Microprocessor Performance Trend



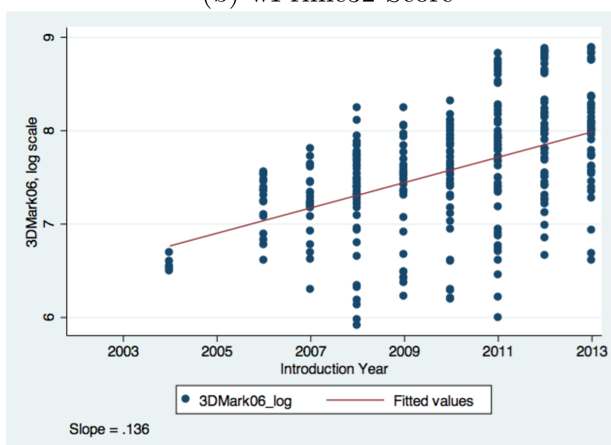
(a) SuperPI1M Score



(b) wPrime32 Score



(c) Synthesized Score



(d) 3DMark06 Score

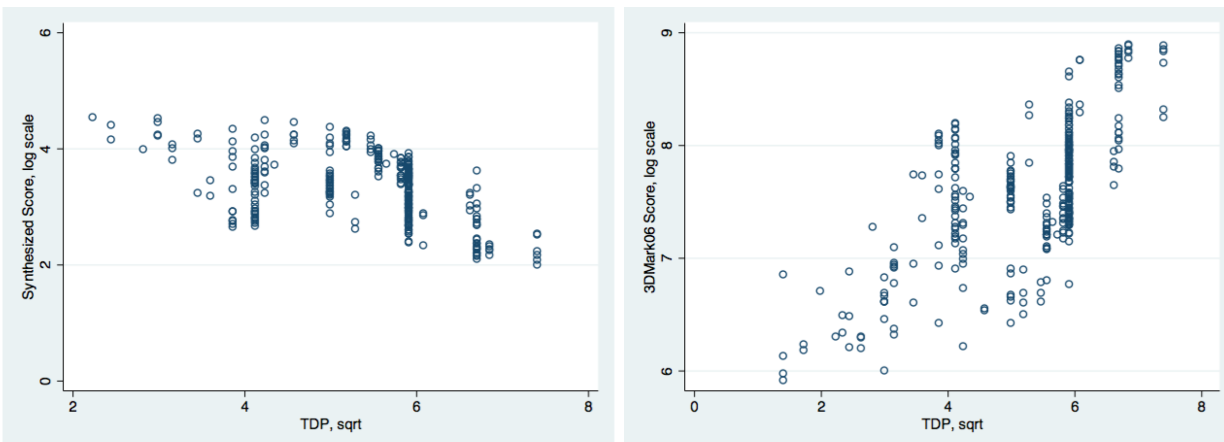
Notes: Trend in (a) microprocessor single-threading ability; (b) multi-threading ability; (c) basic performance; (d) graphics performance.

I plot the benchmark scores against introduction year of the microprocessors in Figure 5.[15] I combine them by constructing a synthesized benchmark, the geometric mean of SuperPI1M and wPrime32, for each microprocessor that I have both SuperPI1M and wPrime32 scores. I use this synthesized score instead of separate SuperPI1M and wPrime32 scores to measure microprocessor basic performance. Comparing the trends in SuperPI1M and wPrime32 scores, the improvement is less steep in SuperPI1M. While both are essential, multi-threading ability becomes more crucial

[15]The sample here includes microprocessors that I can match with benchmark scores, but not necessarily price information, which is different from the sample used in price index estimation.

than single-threading ability for microprocessors since most modern software implement parallel programming models. Of course, by taking the mean, I am assuming equal weights in changes of both scores, which might not be a well-founded synthesis since the importance of multi-threading ability is contingent on how parallelized the software is.[16] I refer to this synthesized benchmark as the "basic" benchmark as its score reflects the microprocessors' performance on basic tasks. Similarly, I refer to 3DMark06 benchmark as the "graphics" benchmark.

Figure 6: Laptop Microprocessor Performance vs. TDP



(a) Basic Performance vs. TDP

(b) Graphics Performance vs. TDP

Notes: The extra low voltage microprocessors (TDP < 5W) are likely netbook microprocessors. They are excluded from my sample of analysis. I include them here for completeness. I take the square roots of TDP to compress its scale for a sensible comparison. The desktop microprocessors used in laptops, which are not included here, have much higher TDP (> 100W.)

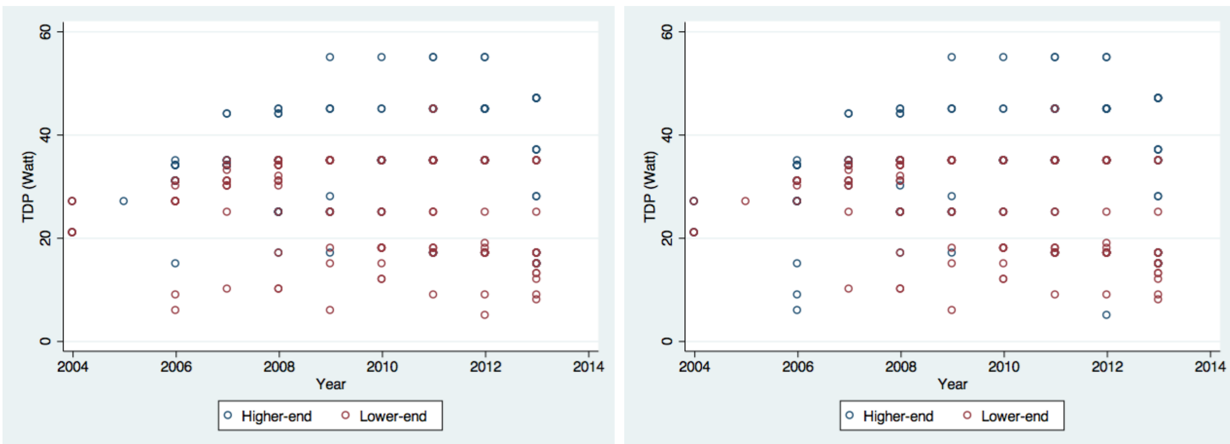
Benchmarks measure performances, the most important quality feature of microprocessors. However, there is a particularly important microprocessor specification that is not reflected in benchmark scores; namely, the Thermal Design Power (TDP), which is the maximum heat the microprocessor generates, such that the cooling system in a laptop is required to dissipate. With a low TDP, the laptop is typically less capable in general. The more powerful microprocessors tend to generate more heat. Thus the cooling systems would require more power to dissipate the heat generated by the microprocessors. Figure 6 plots TDP against benchmark and demonstrates that high TDP tends to correlate with better performance. Figure 7 shows the trend of TDP over the

[16] There are other benchmarks such as PCMark that captures both single-threading and multi-threading abilities. However, I am not able to obtain adequate overlap between price information and those benchmark scores. So I use the geometric mean of SuperPI1M and wPrime32 instead.

past decade, which I examine later in Section 4.4.

In a word, there is no one dimensional metric available for microprocessor quality features. Nonetheless, benchmark scores are at least a consistent indicator for microprocessor performance. I also include TDP in my analysis as an important quality feature.

Figure 7: Laptop Microprocessor TDP Trend



(a) by Basic Performance (b) by Graphics Performance

Notes: Higher-end laptop microprocessors are defined here as the top 50% in terms of benchmark scores, and lower-end laptop microprocessors are the bottom 50%.

3.4 Data Construction: An Example

A brief example in Table 3.4 may help clarify my data construction. Intel introduced Core i7-2640M in 2011Q3, a laptop microprocessor under the Core i7 family. Mid-ranged laptops, such as Lenovo ThinkPad X1 and Macbook Pro assembled in late 2011, installed this microprocessor. I firstly collect technological specifications which help identify microprocessors when linking different sources of data.

For Intel Core i7-2640M, I observe a price quote at $346 on 09/25/2011 from Intel's official website. There are 30 quotes in total until 11/17/2013 from Intel. I also find its price at $238 on 12/01/2012 from the online retail market. There are 4 retail quotes in total until 08/05/2013. In the cases where there are more than one price quote per quarter, I define the quarterly prices as the minimum price of that quarter as mentioned earlier. For Intel prices, I only use the introduction price, the first price in the series for each microprocessor, which is $346 on 09/25/2011. For retail prices, I use the full series of quarterly prices observed. The benchmark scores for Intel Core i7-2640M are on average 11 seconds for SuperPI1M, 15.1 seconds for wPrime32 and 3927 for 3DMark06, collected from NotebookCheck.

The variables that my estimates rely on are quarterly prices, benchmark scores, and TDP. In Section 4, I apply the various hedonic methods introduced in Section 2 and discuss their strengths and weaknesses in the context of my dataset.

Table 1: Data Construction Example

Specifications

Model	Introduction	Lithography(nm)	Clockspeed(MHz)	Turbo Boost(MHz)	# Cores/Threads	TDP(Watt)	Architecture
Intel Core i7-2640M	2011	32	2800	3500	2/4	35	Sandy Bridge

Intel Price

Model	Quarter	Price
Intel Core i7-2640M	2011q3	346
Intel Core i7-2640M	2011q4	346
Intel Core i7-2640M	2012q1	346
Intel Core i7-2640M	2012q2	346
Intel Core i7-2640M	2012q3	346
Intel Core i7-2640M	2012q4	346
Intel Core i7-2640M	2013q1	346
Intel Core i7-2640M	2013q2	346
Intel Core i7-2640M	2013q3	346
Intel Core i7-2640M	2013q4	346

Online Retail Price

Model	Quarter	Price
Intel Core i7-2640M	2012q4	238
Intel Core i7-2640M	2013q1	238
Intel Core i7-2640M	2013q2	188
Intel Core i7-2640M	2013q3	188

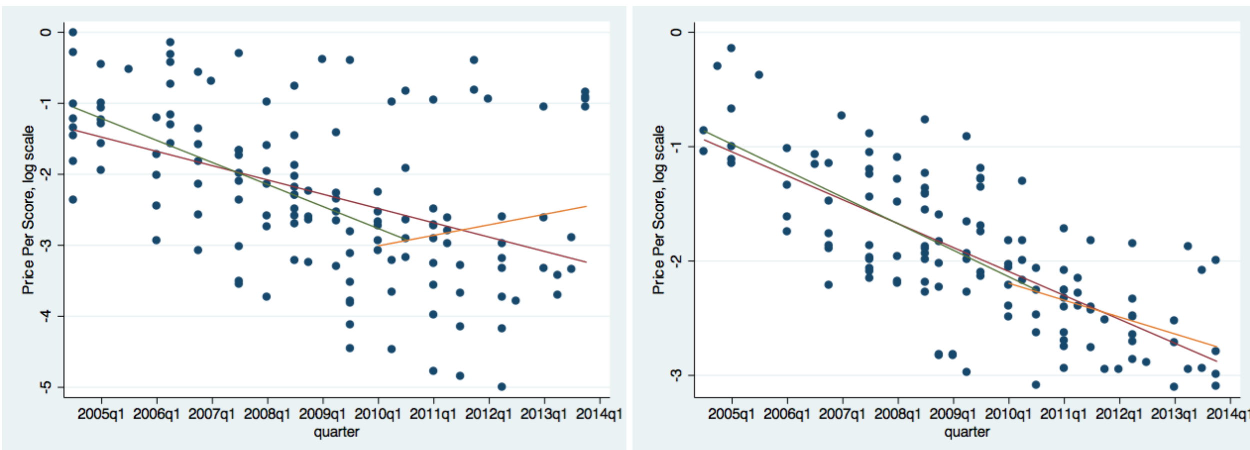
Benchmarks

Model	SuperPI1M	wPrime32	3DMark06
Intel Core i7-2640M	11	15.1	3927

4 Empirical Results

The most intuitive hedonic function form is price per unit of quality. Figure 8 plots the ratio of price to benchmark scores against years. These trends suggest that price per performance declined steadily since 2004, but stagnated since 2010 especially for basic performance as shown in Figure 8(a). These simple ratios fail to address the multi-dimensional nature of microprocessor quality. Besides performance, TDP should also be controlled for as an important quality feature.

Figure 8: Laptop Microprocessor Price Per Performance



(a) Price Per Basic Performance (b) Price Per Graphics Performance

Notes: In (a), price per basic performance = -(Intel initial price) / (gmean of SuperPI 1M and wPrime32 scores). Since the scores are defined in seconds, I flipped the scale so that better performance means a higher (less negative) number. In (b), price per graphics performance = (Intel initial price) / (3DMark06 score).

This concern motivates me to explore richer hedonic functions. Nonetheless, the trends in price per performance hints that starting in 2010, the rate of decline in quality adjusted prices for laptop microprocessor might have indeed slowed down. In order to formally test the hypothesis, I adopt three forms of hedonic functions, from which I develop quality adjusted price indices. I present results from these three estimation methods.

4.1 Dummy-Variable Method

The dummy-variable method pools prices over all periods and imposes one hedonic function. For chip i at time t, I estimate the following:

$$ln(P_{i,t}) = \alpha + \beta ln(benchmark_i) + \gamma ln(TDP_i) + \sum_t \delta_t D_{i,t} + \epsilon_{i,t}.$$

The quality feature variables in Equation 1 are now $ln(benchmark_i)$, log of benchmark scores, and $ln(TDP_i)$, log of TDP Watts. These variables vary across chips but remain constant for a given microprocessor over time. The time dummy variable $D_{i,t}$ is the indicator for whether a price is observed at time t for chip i. As explained in Section 3, I construct a dataset with quarterly prices. The time dummy variables are therefore quarters from 2004 to 2013. To ensure that the initial period is the baseline period, I drop the first quarter dummy variable as noted in Section 2.

The coefficients on the time dummy variables are the main estimates of interest. However, the coefficients on benchmark scores and TDP are also important indicators for the robustness of the models. I show estimates from OLS models in Table 2 for these coefficients.

Table 2: Dummy Variable Method

	I		II	
	Intel Introduction Price		Online Retail Price	
	(a)	(b)	(a)	(b)
ln(benchmark)	-1.545***	1.592***	-1.643***	1.860***
	(0.136)	(0.160)	(0.077)	(0.085)
ln(TDP)	-0.496***	-0.733***	-0.210***	-0.617***
	(0.105)	(0.142)	(0.058)	(0.067)
Observations	136	127	532	447
R-squared	0.716	0.677	0.562	0.585

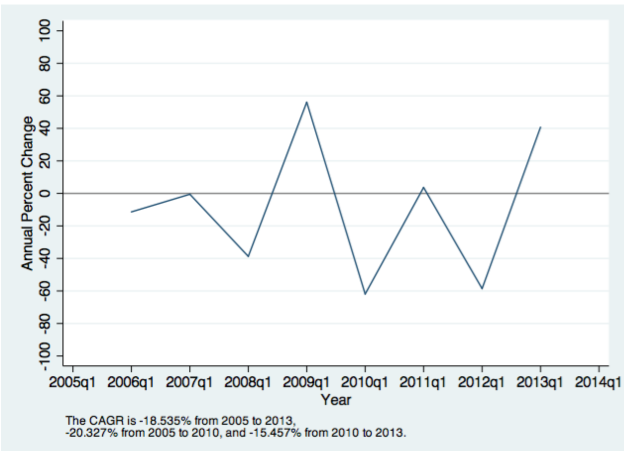
Notes: *** $p < 0.01$, ** $p < 0.05$, * $p < 0.1$. Benchmark variable in subcolumn (a): basic (gmean of SuperPI1M and wPrime32); (b): graphics (3DMark06).

Column I (Intel introduction prices) of Table 2 pools together the introduction prices from Intel ($N = 136$ in I(a) and $N = 127$ in I(b)). Column II (online retail price) changes the price source to online retail prices.

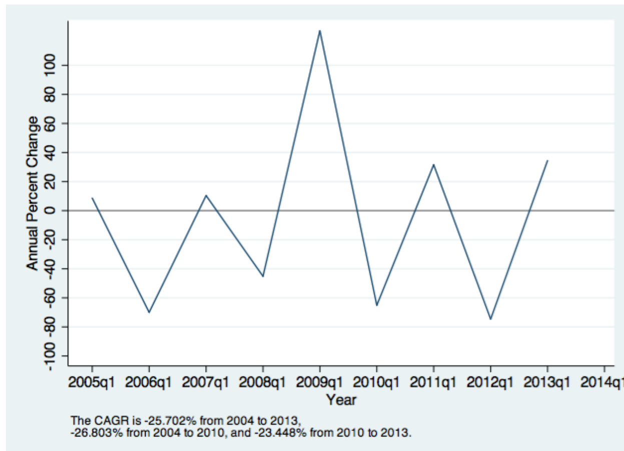
I focus on interpreting the coefficients in column I. The estimates on benchmarks imply that for every 1% reduction in basic score, there is an increase of about 1.6% in price. Similarly, for every 1% increase in graphics score, there is an increase of about 1.9% in price. One way to interpret the magnitudes of these coefficients is the following: over 2004-2013, given the same TDP, to reduce the basic score by half, the price would increase by 80%. To increase the graphics score by half, the price would increase by 95%. The pattern is broadly similar across price sources: price is more responsive to the changes in graphics scores.

The estimates on TDP, on the other hand, are not consistent in magnitude across price sources and benchmarks. In general, we observe a negative sign here because shorter battery life due to higher TDP is undesirable to laptop users given the same performance.

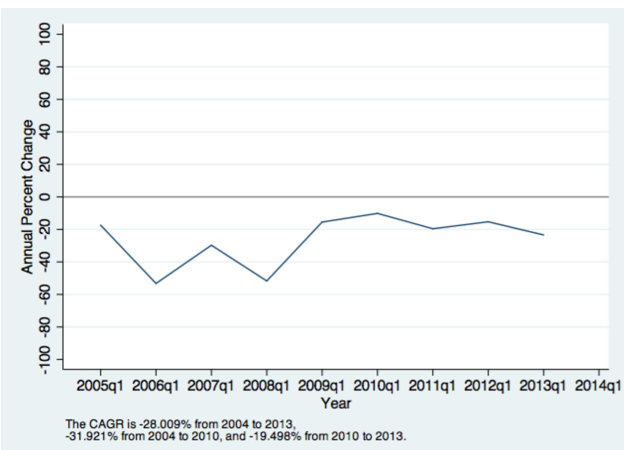
Figure 9: DV Method Quality Adjusted Price Trends



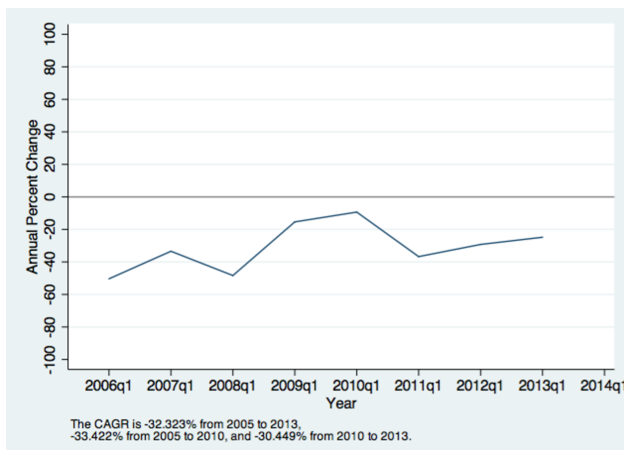
I(a)



I(b)



II(a)



II(b)

Notes: Price sources: Panel I: Intel introduction prices; Panel II: online retail prices. Benchmark variables: Column (a): basic (gmean of SuperPI1M and wPrime32); Column (b): graphics (3DMark06). The start year depends on the availability of data to generate index estimates.

Table A4 in the Appendix documents the δ's and construction of the indices. Since all δ's are relative to the initial period, I can construct annual log difference with simply $\delta_{\text{year t+2 q4}} - \delta_{\text{year t+1 q4}}$ and test its significance. For the sample of intel initial prices, some quarters are missing price quotes. For example, the last quarter observed of year $t+1$ is Q3. To obtain annual log difference, I calculate $\delta_{\text{year t+2 q3}} - \delta_{\text{year t+1 q4}}$ and impute $\delta_{\text{year t+2 q4}} - \delta_{\text{year t+1 q4}}$ by assuming the price decline in Q4 of year $t+1$ is at the average rate over year $t+1$ Q3 to year $t+1$ Q4. Figure 9 compares the annual price declines derived from the indices depending on choice of price sources and benchmark variable. The upper left figure corresponds to the regression results from column I(a) in Table 2, Intel introduction prices as the price source and basic benchmark as the benchmark variable. The rest follows. Details on annual growth rates can be found in Table A5 in the Appendix. I show that the decline from 2010-2013 is indeed significantly smaller than 2004-2010 in magnitudes.

At the outset, DV method constrains the relationship between price change and performance improvement to be the same in all periods. This assumption might not be valid given the ever-changing market of microprocessors. It also constrains the price index to a geometric means formula as discussed in Section 2. I turn to characteristics method, which overcomes these two limitations.

4.2 Characteristics Method

The dummy-variable method constrains coefficients on quality features to be the same over time. This limitation motivates me to adopt the characteristic method for the second part of empirical analysis. The characteristics method, as we examined in Section 2, applies a hedonic function for different time periods separately, rather than pooling together prices from all time periods and assigning time indicators. I apply the characteristics method for each year in my sample. For chip i over adjacent two years $t+1$ and $t+2$, I estimate the following:

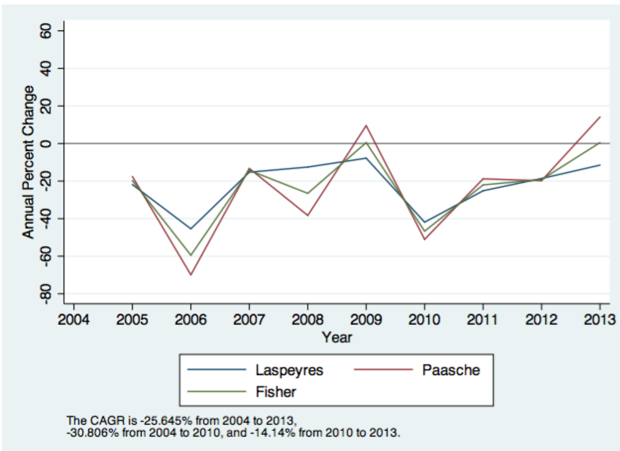
$$h^{t+1}_{char} : ln(P_{i,t+1}) = \alpha_{t+1} + \beta_{t+1} ln(benchmark_i) + \gamma_{t+1} ln(TDP_i) + \epsilon_{i,t+1}$$

$$h^{t+2}_{char} : ln(P_{i,t+2}) = \alpha_{t+2} + \beta_{t+2} ln(benchmark_i) + \gamma_{t+2} ln(TDP_i) + \epsilon_{i,t+2}$$

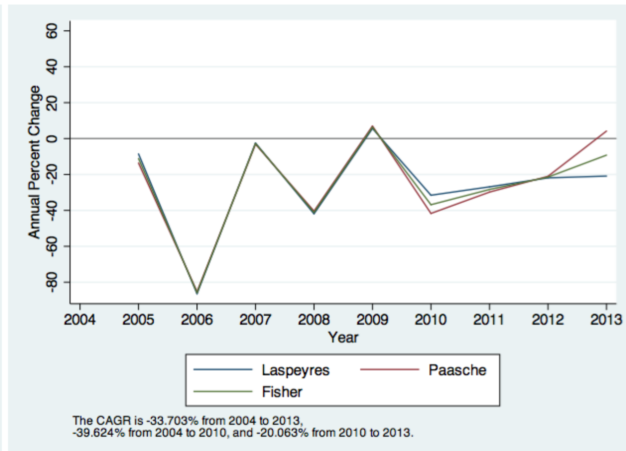
I construct and describe price indices from the regression results. For now I focus on the coefficients on benchmark scores for Intel and online retail prices. I show estimates from OLS models in Table A6 in the Appendix. I firstly note that the coefficients on the characteristics terms indeed vary over years. Generally, the magnitude of coefficients on basic benchmark is larger than graphics benchmark, which is consistent with results from the DV method.

I report the predicted average characteristics prices and Laspeyres, Paasche and Fisher indices in Table A7 in the Appendix. Figure 10 presents the annual growth estimated by the characteristics method. Details on annual growth rates can be found in Table A8 in the Appendix. The overall trend conveys a similar message to that in the DV results: quality adjusted prices decline rapidly since 2004, but the decline has slowed down since 2010.

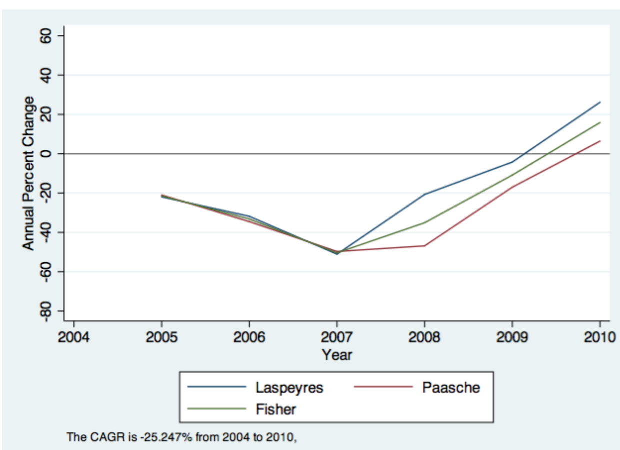
Figure 10: Characteristics Method Quality Adjusted Price Trends



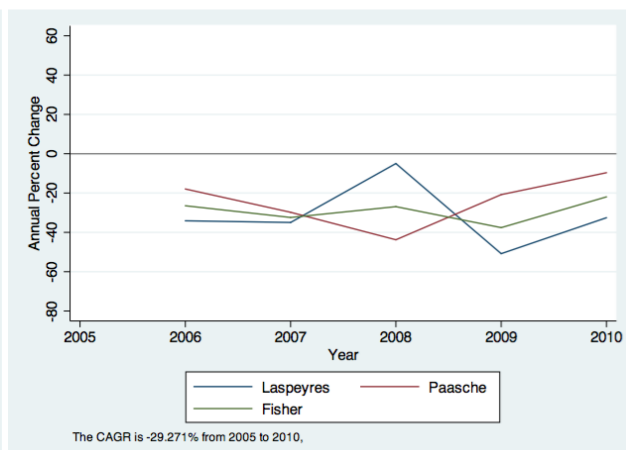
I(a)



I(b)



II(a)



II(b)

Notes: CAGR calculated using Fisher index. I truncate the sample for online retail prices to end in 2010. When constructing the dataset, I append prices from a different source for 2011-2013 to the previous prices. The latter source covers a rather different microprocessor population, which results in noncomparable characteristic function estimates.

The first advantage of the characteristics method is that it constrains the coefficients on quality features as little as possible. The DV method requires that the coefficients be unchanged over all time periods. The characteristics method is not subject to this criticism. It uses one hedonic function for each of the periods included in the price index, not a single hedonic function for all periods, over which characteristics prices can change. However, what matters is the price index, not the coefficients themselves. As we compare the indices from these two methods, the overall similarity suggests that empirically these two methods do not differ by much given my dataset. That said, when there are too few observations within a certain year, or the samples from two consecutive years overlap poorly, the characteristics method could generate unreliable estimates.[17] The second advantage is characteristics method separates the index formulation from the hedonic function. The hedonic function depends on the empirical relationship between the prices and qualities given that it is highly reduced form. The index formula depends on the usual theoretical conditions from index theory, not on the statistical relation established empirically by the hedonic function. The DV method implies an index formulation, which is the ratio between geometric means in quality adjusted prices, which might not be the desired price index formula. The price index calculated using the characteristics method does not assume a connection between hedonic functional form and index number functional form.

The downside of characteristics method is that I omit quarterly differences in prices. By pooling prices on yearly basis, I essentially take a weighted average of prices within a year. Given that prices for microprocessor change rapidly, we might not want to ignore the quarterly differences in prices. Also, the variation in sample size and population introduces large volatility to the estimations.

Thus I return to a tweaked version of dummy-variable method. Instead of inserting time dummy variables, I add a linear time trend over quarters in adjacent years. Such implementation constrains the coefficients as little as possible, and at the same time permits reasonable sample size for each hedonic function.

[17]This dilemma happens to the online retail price sample. For 2010 and 2011, the microprocessor population changes drastically and results in noncomparable characteristic function estimates.

4.3 Adjacent-Year Method

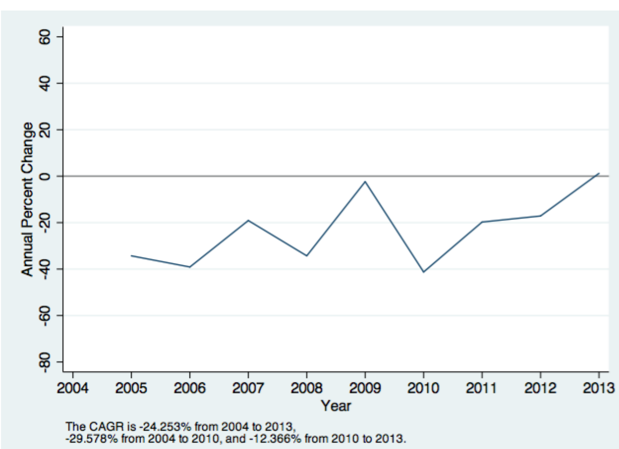
I present my third and last empirical estimation method. It is referred to as the adjacent-year method. For two overlapping years, for chip i at quarter t, I estimate the following:

$$ln(P_{i,t}) = \alpha + \beta ln(benchmark_i) + \gamma ln(TDP_i) + \delta t_{i,t} + \epsilon_{i,t}$$

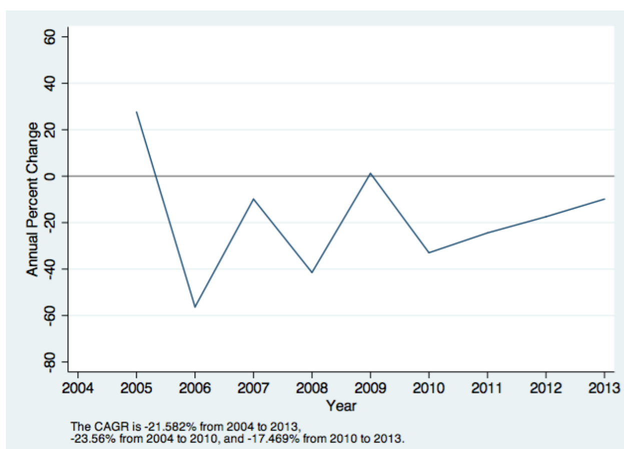
The adjacent-year method differs from the dummy variable method in two aspects: 1) rather than pooling prices from all periods, I limit the sample to two adjacent year for each hedonic function. Therefore instead of fitting one hedonic function on the entire dataset (2004-2013), I allow a unique hedonic function for each two-year window. Thus there are nine functions in total. 2) I substitute the time dummy variable with a linear time trend. The advantage is that the coefficient on the time trend, δ, is already the average quarterly growth rate, which I can easily convert to annual growth rate. This shortcut saves the imputation for annual growth rate in DV method, which is especially helpful dealing with Intel first price dataset.

In the Appendix, Table A9 lists the coefficient estimates on benchmark variables. As expected, the estimates are more stable over years compared with the characteristic method. However, the pattern is generally similar: the magnitude for coefficients on graphics benchmark is larger than that of basic benchmark.

Figure 11: Adjacent-Period Method Quality Adjusted Price Trends

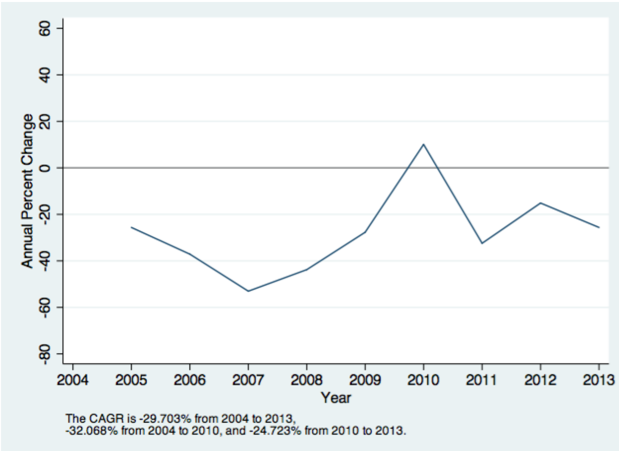


I(a)

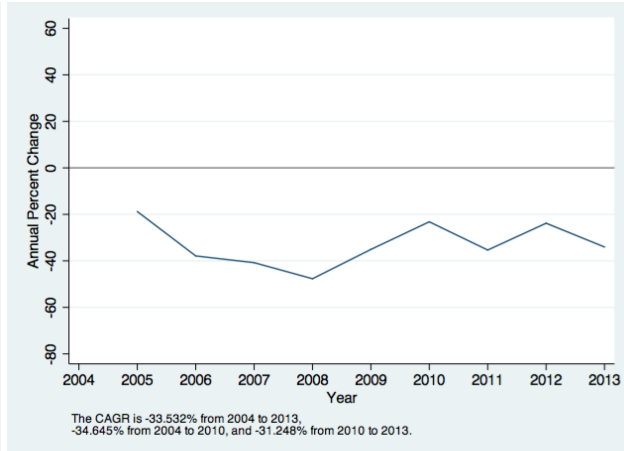


I(b)

Table A10 in the Appendix shows the annual growth rates calculated from δ's. Figure 11



II(a)



II(b)

plots the annual percent change in the estimated quality adjusted price indices. Again, despite the sharp price decline over the past ten years, there is a noticeable slowdown since 2010.

4.4 Potential Causes of Slowdown

Table 3 reports the CAGR of quality adjusted price calculated using aforementioned hedonic methods. Over 2004-2013, the quality adjusted price decline on average from -19% to -34%, with the mid-range at -27%. The basic performance adjusted price decline ranges from -19% to -30%, with the mid-range at -26%. The graphics performance adjusted price decline ranges from -22% to -34%, with the mid-range at -28%.

Table 3: Summary of Compound Annual Growth Rates

CAGR for 2004-2013

	I		II	
	(a)	(b)	(a)	(b)
DV Method	-19	-26	-28	-32
Characteristics Method	-26	-34	n/a	n/a
Adjacent-Period Method	-24	-22	-30	-34

CAGR for 2004-2010 and 2010-2013

	DV Method				Characteristics Method				Adjacent-Period Method			
	I		II		I		II		I		II	
	(a)	(b)	(a)	(b)	(a)	(b)	(a)	(b)	(a)	(b)	(a)	(b)
2004-2010	-20	-27	-32	-33	-31	-40	-25	-29	-30	-24	-32	-35
2010-2013	-15	-23	-19	-30	-14	-20	n/a	n/a	-12	-17	-25	-31

<u>Notes:</u> Price sources: Panel I: Intel introduction prices; Panel II: online retail prices. Benchmark variables: Column (a): basic (gmean of SuperPI1M and wPrime32); Column (b): graphics (3DMark06). Fisher index is reported here for the characteristic method results.

I summarize CAGR for 2004-2010 and 2010-2013 as well. The comparison between 2004-2010 and 2010-2013 in Table 3 recapitulates the slowdown in rates of price decline since 2010. All three hedonic methods (DV method, characteristics method, and adjacent-period method) have delivered consistent estimates across choices of prices sources and benchmark variables. By that I mean given a hedonic method, if we condition on price source and benchmark variable, the average price change in 2010-2013 is consistently milder compared with that in 2004-2010. That is, the values in row “2010-2013” are smaller in magnitude than those in row “2004-2010”. For example, using the adjacent-period method, I estimate the average price change to be -30% in 2004-2010 in Intel introduction prices adjusting for basic performance. With the identical regression set-up, I estimate the average price change to be -12% in 2010-2013, which is smaller in magnitude compared with 2004-2010. Such pattern is uniform regardless of the set-up.

There is also a consistent pattern in price trends conditioned on the choice of benchmark variables. Comparing estimates in subcolumns (a) and (b) under the same price source, the graphics performance adjusted price indices exhibit larger decline than the basic performance adjust price indices in most cases. This pattern might imply a slower improvement in basic performance, which I confirm later with empirical evidence.

Overall, especially after 2010, online retail prices decline at a faster rate than Intel introduction prices. Comparing subcolumn I and II with the same benchmark variable, the online retail prices experience faster drop than the Intel introduction prices. As mentioned in Section 3, the online retail market represent a higher-end segment and display more variations in prices. Suppose pricing schemes are similar across the product line, then the slower drop in Intel price may imply smaller price adjustment for new offerings of microprocessors since 2010. Later I relate this observation to potential changes in pricing strategies by Intel.

I conclude that my estimates of hedonic quality adjusted prices show sharp decline in the order of about -20% to -30% annually, but represent a slower decline at about -15% to -25% since 2010. By definition, the hedonic function is the expectation of price conditioned on quality features. The changes in hedonic quality adjusted prices are therefore changes in expectation of price consisting of marginal cost and mark-up given constant quality. So we can attribute the slowdown in quality adjusted price potentially to either or both of the following two broad channels: 1) smaller improvement in quality which results in smaller price declines, and consequently slower quality adjusted price declines 2) different pricing strategies with faster increase in marginal cost and/or mark-up.

4.4.1 Quality Improvement

To test for the possibility of slower improvement in quality, I repeat the growth rates analysis on benchmark scores.[18] For the trends displayed in Figure 5, I present average annual growth rates in Table 4 I, separated by year 2010. This simple analysis corroborates the trends in quality adjusted prices: overall substantial improvement with year 2010 as the turning point.

[18] The growth rates are CAGR, which can be imprecise since the progression on the benchmark scores might not be linear. For example, a 20% increase from 500 to 600 might imply a larger improvement than a 20% increase from 100 to 120.

I firstly caution against comparing the growth rates across benchmarks. A direct comparison between growth rates would imply the two benchmarks have similar scoring paradigms. That is, a 10% decrease in basic benchmark scores is as attainable as a 10% increase in graphics benchmark scores. Such generalization is flawed. Therefore we may only compare the changes in improvement rates for the two benchmarks respectively.

Table 4: Laptop Microprocessor Quality Improvement

I. Performance Improvement

	Basic Performance			Graphics Performance		
	(1)	(2)	(3)	(1)	(2)	(3)
2004-2010	-21%	-16%	-9%	20%	10%	4%
2010-2013	-10%	-5%	-2%	6%	10%	19%
2004-2013	-17%	-12%	-7%	18%	11%	6%

II. Performance per $TDP^{1/2}$ Improvement

	Basic Performance			Graphics Performance		
	(1)	(2)	(3)	(1)	(2)	(3)
2004-2010	-24%	-18%	-7%	19%	13%	10%
2010-2013	-9%	3%	0%	10%	17%	28%
2004-2013	-19%	-12%	-5%	17%	13%	9%

III. Decomposing Basic Performance

	Top			Median			Bottom		
	Basic	(a)	(b)	Basic	(a)	(b)	Basic	(a)	(b)
2004-2010	-21%	-16%	-25%	-16%	-11%	-20%	-9%	-5%	-14%
2010-2013	-10%	-7%	-14%	-5%	-8%	-6%	-2%	1%	-3%
2004-2013	-17%	-13%	-21%	-12%	-10%	-15%	-7%	-3%	-11%

Notes: Categorization in Panel I and II are (1) high-end (2) mid-range (3) lower-end.
Improvement in the high-end segment of the microprocessors is defined as the growth rates in the geometric mean scores of the top 25% (≤75% percentile score for basic or ≥25% percentile score for graphics) benchmark scores; improvement in the mid-range segment is defined as the growth rates in the geometric mean scores of the middle 50%, i.e. (25%,75%] benchmark scores; improvement in the lower-end segment is defined as the growth rates in the geometric mean scores of the bottom 25% (basic: >25% percentile score; graphics: <75% percentile score) benchmark scores.
Benchmark variables in Panel III are (a) SuperPI1M and (b) wPrime32.

Basic performance, measured by the geometric mean of SuperPI1M and wPrime32 scores, improved at a slower rate 2010-2013 (-16%) than 2004-2010 (-5%) for the mid-range laptop microprocessors. Graphics performance, measured by the 3DMark06 scores, improved at the same rate 2010-2013 (10%) as 2004-2010 (10%) for the mid-range laptop microprocessors. This difference is consistent with the slower decline in basic performance adjusted price.[19]

We can further dissect the performance trends by incorporating TDP. I factor in TDP by dividing the benchmark scores with the squared roots of TDP. Shown in Table 4 II, after controlling for TDP, mid-range graphics performance improved at a faster rate post-2010 (17%) than pre-2010 (13%). Since the graphics creation tends to generate more heat than ordinary tasks, the improvement in graphics performance is often at the cost of higher TDP. It is particularly meaningful for laptop microprocessors to improve graphics performance without raising TDP. So I conclude that the improvement in graphic performance did not slow down post-2010, after controlling for power consumption.

By contrast, controlling for TDP seems to exacerbate the slowdown in basic performance improvement. I decompose the basic performance into single-threading and multi-threading performance to analyze the trends more thoroughly. In Table 4 III, I observe that across the laptop microprocessor product line, i.e. high-end segment vs. lower-end segment, there is a noticeable slowdown in both SuperPI1M and wPrime32 improvement (comparing rows of "2004-2010" with "2010-2013" under sub columns (a) and (b) within each segment.)

Furthermore, there is evidence that basic performance improvement is at a slower rate for laptop microprocessors than for desktop microprocessors.[20] Similar decomposition of growth rates in basic performance on desktop benchmark scores shows a contrasting pattern in Table 5. Indeed, the rate of basic performance improvement in desktop microprocessors persisted throughout 2004-2013, at -21% pre-2010 and -20% post-2010 for the mid-range microprocessors (Column (2) of Table 5, comparing row "2004-2010" with row "2010-2013" under "Basic") .[21] Similarly to laptop microprocessors, the single-threading performance as measured by SuperPI1M score shows stagnation overall. In response to more and more parallelized software, Intel shifted its research focus from single-threading to multi-threading performance of microprocessors as early as 2005. Therefore I focus on the multi-threading performance as measured by wPrime32 score.

[19]The shift in architecture might explain the disparity. Starting in 2010, the microprocessor industry began to integrate GPU into microprocessor, which substantially increased graphics performance.

[20]I retrieve the benchmark information for desktop microprocessor from YouCPU, which retrieves average benchmark scores from benchmark websites similarly as NotebookCheck.

[21] Byrne et al. [2014] suggest that the performance of Intel's desktop microprocessors improved roughly 30 percent per year on average from 2001 to 2012, based on benchmark scores from System Performance Evaluation Corporation (SPEC). Since SPEC is a more comprehensive benchmark, my estimates for improvement in desktop are likely to be different.

Table 5: Desktop Microprocessor Quality Improvement

	(1)			(2)			(3)		
	Basic	(a)	(b)	Basic	(a)	(b)	Basic	(a)	(b)
2004-2010	-27%	-15%	-33%	-21%	-10%	-24%	-11%	0%	-13%
2010-2013	-13%	-8%	-15%	-20%	-10%	-22%	-18%	-10%	-25%
2004-2013	-23%	-13%	-28%	-21%	-10%	-23%	-14%	-3%	-17%

Notes: Categorization in Panel I and II are (1) high-end (2) mid-range (3) lower-end. Benchmark variables in subcolumns are (a) SuperPI1M and (b) wPrime32. This desktop sample includes quite a few AMD microprocessors. I include them here as a complete comparison with the laptop microprocessor benchmark information, but I note that the difference in AMD and Intel architectures can cause various patterns in quality improvement over time.

Across the product line of desktop and laptop microprocessors, the patterns are similar on improvement rate between high-end and lower-end for 2004-2010. For 2004-2010, performance improves rapidly for the entire product line of desktop and laptop microprocessors, especially for higher-end microprocessors. For 2010-2013, I notice a different pattern among laptop microprocessors: the mid-range and lower-end have slowed the improvement rate substantially. The improvement rates are, by contrast, rather consistent among higher-end, mid-range and lower-end desktop microprocessor. The contrasting performance might signal differentiated product introduction strategies by manufacturers for laptop microprocessors. Manufacturers consistently developed and introduced better microprocessors than the previous year across the product line each year for 2004-2010. This strategy is studied in detail by Nosko [2010]. However, after 2010, it seemed that the development effort was focused on the high-end segment of the microprocessors. Then the improvement in the high-end segment gradually extends to the entire product line. The continued improvement in the lower-end segment could just be a "waterfalling" effect from the high-end segment, which perhaps explain the slowdown in the overall improvement in laptop basic performance.

For the slowdown in quality improvement, I present two potential explanations based on the technological progress in the microprocessor industry outline by Aizcorbe and Kortum [2005]. The technological progress in microprocessors is largely driven by innovations in the upstream semiconductor equipment industry. Semiconductor equipment firms like Nikon and Canon invent new generations of capital equipment which allows microprocessor firms like Intel and AMD to manufacture smaller transistors, enabling them to make higher quality microprocessors. As shown in Table 6, the current size of transistor is 22nm introduced in 2012. For past decades, innovations

Table 6: Intel Microprocessor Family Introduction Timeline

	2003	2006	2008	2010	2012
Family	Pentium M	Core 2 Duo	Core	2nd gen. Core	3rd gen. Core
Clockspeed	1.7GHz	2.66GHz	2.4GHz	3.8GHz	2.9GHz
# of Transistors	55 million	291 million	410 million	1.16 billion	1.4billion
Lithography	90nm	65nm	45nm	32nm	22nm
Architecture	P6	Core	Nehalem	Westmere	Sandy/Ivy Bridge

Notes: This timeline summarizes the introduction of microprocessor families analyzed in this paper. More details are available at Intel Chips Timeline (`http://www.intel.com/content/www/us/en/history/history-intel-chips-timeline-poster.html`.) The laptop microprocessors architectures is Clarksfield under Nehalem and Arrandale under Westmere.

in the upstream semiconductor equipment industry has allowed a shrinkage in size by a factor of roughly 0.7 every 2-3 years, which would shrink the transistor area by half. The ability to constantly halve the transistor sizes fulfills the Moore's law, a statement made in Moore [1965] that the number of transistors on a microprocessor doubles every two years. As transistor size reaches infinitesimal, the manufacturing becomes more challenging and the fixed-cost skyrockets. "When Moore's Law ends, it will be economics that stops it, not physics." Robert Colwell, director of the Microsystems Technology Office at the Defense Advanced Research Projects Agency (DARPA), emphasized in August, 2013.[22] The increasing cost might have already stopped Moore's law. Somewhere in between 2010 and 2012, as shown in Table 6, the number of transistor for the 3^{rd} generation Core ceased to increase as fast as before. Consequently there is not as much momentum for improvement directly from the upstream semiconductor equipment industry as before.

The second explanation lies within the microprocessor industry. Although the industry has progressed at the speed predicted by the Moore's law, it has not been able to efficiently transform the addition of transistors to quality improvement as suggested by Pillai [2013]. With his model of technological progress in the microprocessor industry, Pillai [2013] suggests the slowdown in quality was mainly caused by a decrease in the efficiency with which manufactures were able to use the innovations generated by the upstream semiconductor equipment industry. For example, even though the area of transistor was halved, clockspeed did not improve from 2008 to 2010 by as much (between Nehalem and Westmere architectures) as shown in Table 6.[23] Borkar and Chien

[22]See "The Chip Design Game at the End of Moore's Law." The original presentation is available at `http://www.hotchips.org/wp-content/uploads/hc_archives/hc25/HC25.15-keynote1-Chipdesign-epub/HC25.26.190-Keynote1-ChipDesignGame-Colwell-DARPA.pdf`. It was presented at the Hot Chips Conference at Stanford University.

[23]In general, comparison on clockspeed of microprocessors from different architectures is rather invalid because

[2011] explain the difficulty in a more technical context. I highlight here the challenge specific to laptop microprocessors. Additional transistors generate more heat, which demands extra power to dissipate the heat. Due to its portable nature, power consumption is a stricter constraint on laptop microprocessor design than for desktop microprocessors. Energy efficiency, on the other hand as suggested by Figure 7, has not been consistently improving. This constraint might also explain smaller improvement in laptop microprocessor quality than desktop. Nonetheless, as we saw earlier, graphics performance seems to be improving without the cost of raising TDP. The laptop microprocessor industry might be experiencing and energy efficiency breakthrough.

Additionally, the unique market structure might affect the innovation rate and therefore quality improvement. Goettler and Gordon [2011] find that innovation is higher with Intel as a monopolist than with an Intel-AMD duopolist over the span of 1993-2004. The microprocessor market has shifted to predominantly an Intel monopoly rather than an Intel-AMD duopoly. The effect of competition on innovations in the microprocessor market could have changed as well. Two kinds of competition drive innovation: competition between manufactures for the technological frontier and competition with the current quality to induce consumers to upgrade. Duopolists face both, whereas a monopolist faces only the latter. Moreover, rapid market growth from first-time laptop buyers especially in 2000s might have reduced innovation incentives for Intel who could exploit their demand. To boost demand from laptop replacements, Intel needs to widen the quality gap between currently owned and new offerings for second-time buyers. Demand from first-time buyers may not be as responsive to quality changes as second-time buyers. Potentially the first-time buyers are more attuned to graphics performance and energy efficiency than basic performance. Therefore Intel might not face much competition with the current quality of laptop microprocessor, especially in terms of basic performance, which might have exacerbated the slower rate of quality improvement.

---

the length of pipeline varies. That is, the amount of work done per clock cycle is different for each architecture. However, since Westmere is the “tick” following Nahalem, I assume the main quality improvement is mostly from the shrinkage in transistor size, not changes in architecture. (Intel’s “tick-tock” model means every “tick” is a shrinkage of transistor size of the previous architecture and every ”tock” is a new architecture. Therefore the comparison of clockspeed is not completely groundless for Nahalem and Westmere.

4.4.2 Pricing Strategies

The smaller quality improvement alone might not fully explain the slowdown in price decline. Potential changes in pricing strategies might have also contributed to the slowdown

I rely on previous literature and qualitative evidence to discuss the possibility of changes in pricing strategies.

The marginal cost might have increased at a faster pace recently. As explained in Pillai [2013], larger microprocessors are associated with higher marginal costs of manufacturing because larger sizes increase the fraction of microprocessors that are defective. The higher defect rate arises from contamination by dust particles in the manufacturing process. Larger microprocessors have a higher probability of being contaminated. According to the Poisson yield equation used in the industry, the defection rate grows exponentially with the area of a microprocessor.

To improve quality, it might require fitting more transistors or a more complex design on a microprocessor and therefore increase its size. Although there is no ample information on sizes of microprocessors in my dataset to verify this hypothesis, it is possible that the microprocessor size has increased lately as more and more new features are added to its architecture: multi cores in 2005, memory controller in 2008, and GPU in 2010.[24] These additions might have enlarged the size and escalated the marginal cost of microprocessors.

The mark-up might have risen to justify the skyrocketing fixed cost in the industry. In an investor meeting late 2013, Intel's chief financial office Stacy Smith indicated that Intel's factories were operating at less than 80% of capacity.[25] Each manufacturing plant requires a staggering amount of investment and yet some of them have remained idle due to lack in volume. Perhaps the overcapacity has in part induced larger mark-ups.

The mark-up varies with demand and the elasticity of demand. Since microprocessors are largely durable goods, the replacement decision differs from the first-time buying decision. As the microprocessor market matures, replacement purchases dominates the proportion of sales. With a dynamic model of consumer replacement cycles, Gordon [2009] finds significant variations in the distributions of replacement cycles across consumers over the period of 1993 to 2004. On

[24] See the Appendix for a comparison of architectures in Figure A3.

[25] The original presentation is available at `http://mindspace.ru/wp-content/uploads/2013/12/INTC_investor_meeting_2013_IM_Smith.pdf`

average, consumers replace their existing processors about every 3.3 years. The high-end consumers have a shorter replacement cycle than the lower-end consumers. It would be valuable to perform similar analysis on the replacement cycle since 2004 for the laptop microprocessor market. Possibly replacement cycles have been stretched longer and consumers have become less sensitive to prices over time considering the introduction of tablets.

Lastly, Intel's pricing scheme for new offerings of microprocessors could have changed since 2010. My estimates indicate quality adjusted price decline slower in Intel introduction prices than in online retail prices after 2010. This might suggest that for a mid-ranged microprocessor introduced in the current period, compared with a mid-ranged microprocessor introduced in the last period, Intel does not adjust downward the price as much as the retail market would have equilibrated. If the demand elasticity has indeed decreased since 2010, then Intel would be able to extract more revenue from the less price-sensitive consumers.

5 Estimation Issues

Sources of possible errors in my estimates are: measurement error, selection bias and flaws in hedonic methods.

5.1 Measurement Errors

There could be measurement errors in microprocessor prices. Intel prices, even the introduction prices, might not be actual transaction prices. Intel reports prices for microprocessors offered to distributors and original equipment manufactures (OEM) when they purchase in lots of 1,000. Large buyers might get discounts based on the size of their purchases. I cannot rule out the possibility that the actual transaction price might be lower than the list price. This discrepancy would not cause bias in my analysis as long as the discounts do not vary over time or model. However, this assumption is nearly impossible to verify.

Similarly, the retail market price might cover more than a microprocessor. Although I call this source the online retail prices, most of the prices listed are for the OEM versions of the microprocessors, also referred to as tray prices. The OEM microprocessors usually do not come equipped with a heat sink or fan. The actual retail prices are higher since they generally include the longer warranties, heat sinks and fan which OEM prices do not. Therefore only the OEM prices are comparable to those on the Intel website. There might be a few actual retail prices, and I assume the additional charge is small.

5.2 Selection Bias

Benchmark scores, on the other hand, pose mostly selection bias on my sample. 3DMark is a well-known benchmark and gets quoted in microprocessor reviews often. But the other two benchmarks I choose, SuperPI1M and wPrime32, are not administered by private companies or official organizations. It is the community of computer hobbyists that maintains and updates the score charts for microprocessors. The hobbyists make purchase choices based on the benchmark scores, which is probably different from how an average consumer makes purchase choices. As I mentioned in Section 3, only the "popular" microprocessors get benchmarked, and the "popular" among hobbyists might be the higher performing ones. Thus my sample is likely biased upward

Table 7: Coverage on Intel Laptop Microprocessors 2004-2013

Before Matching		After Matching			
Family	Counts		SuperPI1M	wPrime32	3DMark06
Celeron	54	Celeron	17	14	13
Celeron Dual Core	2	Celeron Dual-Core	2	0	2
Celeron M	50	Celeron M	16	14	5
Core 2 Duo	45	Core 2 Duo	37	39	36
Core 2 Extreme	5	Core 2 Extreme	5	5	5
Core 2 Quad	2	Core 2 Quad	2	2	2
Core 2 Solo	4	Core 2 Solo	2	1	2
Core Duo	13	Core Duo	9	6	6
Core Solo	5	Core Solo	5	3	1
Core i5	25	Core i5	14	14	14
Core i7	50	Core i7	33	32	32
Pentium M	35	Pentium M	15	14	9
Total	290	Total	157	144	127

in terms of quality features compared to the market. Therefore, the issues of potential selection bias are twofold: 1) how many microprocessors are benchmarked; 2) how representative are the "popular" benchmarked microprocessors of the microprocessor market.

Table 7 hints at the share of microprocessors benchmarked. On the left is the sample of Intel laptop microprocessors with price information. I assume this sample covers most of the Intel laptop microprocessors marketed in 2004-2013. On the right is the sample of Intel laptop microprocessors that I obtained benchmark scores for. While about half microprocessors are benchmarked overall, a few families such as "Celeron" and "Pentium M" are subjected to major selections. However, I note that the selectios in these families might not pose a threat to the reliability of my estimates if their microprocessors are not the mainstream microprocessors during their marketing period.

To investigate how representative the benchmarked microprocessors are, I look for anecdotal data from the online retail market on which microprocessors are the best sellers and verify that they are in the benchmarked sample. I attempted to use records on Amazon as evidence for popularity. Out of the 40 bestseller microprocessors from a Amazon search in April, 2014, my Intel price dataset has benchmarks for 21 of them whereas the online retail dataset has only 8.[26] However, since the Amazon ratings are self-reported and accumulative, it is difficult to differentiate the mainstream microprocessors. To illustrate this point, I note that one of the bestsellers, Intel Core

[26] The 40 bestsellers are: Celeron M (1), Intel Core 2 Duo (16), Intel Core 2 Extreme (1), Intel Core 2 Quad (1), Intel Core i3 (3), Intel Core i5 (6), Intel Core i7 (6), Pentium 4 (2), AMD (3) microprocessors. The list is obtained on April 20, 2014.

2 Duo T8300, is sold at \$38, compared with its introduction price of \$241 in 2008. The reviews on the product suggest that the buyers refurbished their obsolete laptops with this microprocessor. Another bestselling microprocessor, Intel Core i7-2820QM, is sold at \$498, compared with its introduction price of \$568 in 2011. The reviews on the product suggest that the buyers upgraded their laptops which are already in fine condition. These two microprocessors are in different segments of the market at present. Being in the sample of Amazon bestseller microprocessors does not prove that they are the prevailing microprocessors of their marking periods. For future study, I hope to obtain more accurate reports on the annual bestsellers in the laptop microprocessor market to answer this question.

5.3 Flaws in Hedonic Method

Estimating hedonic indices involves two parts: selection of hedonic functional forms and index number formula. As shown in Section 2, a hedonic function is only a statistical relation between the prices of a sample of goods and their qualities. In turn, these qualities are both outputs for producers and commodities for consumers, which is the underlying assumption for the hedonic method. Producers manufacture a bundle of quality features. Similarly, consumers make their purchase decision based on the bundle of quality features.

To formulate a hedonic function, one needs to select variables that are representative of quality features. Hedonic method is therefore prone to criticism for the "arbitrary" selection of quality variables. The two features I selected, benchmark scores and TDP, certainly cannot encompass all abilities of microprocessors. To determine the quality features that consumers value the most, we certainly need estimates of the distributions of preferences over quality features. Nonetheless, based on my research I argue that these two features are the most basic ones of laptop microprocessors, and average consumers are most likely to factor these two features in when making purchase decisions.

I emphasize that hedonic functions embody both supply and demand sides of the market. The coefficients are merely elasticities between price and quality. In perfect competition, the coefficients represent the marginal cost of improving quality. But as we learned in Section 3, the microprocessor market is far from perfectly competitive. Thus, the hedonic functions also include the mark-up, and we may only check the plausibility and comment on the pattern of these coefficients. Consider, for example, since multi-threading ability is desired by the consumers and costly to producers, we would be surprised to observe a negative relationship between them in the hedonic function. But we may not draw any quantitative conclusion on producer cost or user value.

Regarding index formulation, criticism centers around the unweighted index implied by hedonic functions. This criticism, however, is based on some confusion about hedonic functions. Rather than incorporating market share as weights, hedonic functions weight the indices by qualities. Therefore it is not entirely unweighted. Not weighting by market share indeed gives "one vote per model" unfairly. However, using solely market share weights in price index is erroneous as well. For example, the market share for a microprocessor can be small because it supplies a market niche or it is a market failure. For future study, I hope to approximate market share by the number of processors per family or revenue shares per family.

From a practical point of view, what is the best practice for calculating price indices? Price indices should be timely and accurate. Clearly hedonic methods fail the first criteria. The quarterly index is highly volatile and unstable, especially when data are limited. Matched model methods, on the other hand, can construct indices upon availability of prices. Nonetheless hedonic methods might be better attuned to quality changes for the microprocessor industry, in which new goods bias is prominent. My empirical estimates show that hedonic methods control for quality more accurately than matched model method.

There are certainly other considerations in terms of price index choice for statistical agencies. Statistical agencies are accountable for the authority in their price index. An anecdote is that during the climax of the antitrust lawsuit *Federal Trade Commission v. Intel*, Intel requested a subpoena to examine microprocessor price indices from BLS. Intel claimed that the biased price indices from the BLS constituted evidence for the allegations of illegal anticompetitive tactics.[27] Compared with the hedonic method, the matched-model method might offer a more definite explanation on the price index formulation in such a situation. While the hedonic method can take on different functional forms, the matched-model method involves no variance from different forms of estimation methods. Therefore statistical agencies might prefer the matched-model method for its straightforwardness. However, the possibility of biases in matched-model indices suggests that statistical agencies should

[27] Intel complained the BLS PPI fell too fast over 2000-2009. I was unable to access more details from either side regarding Intel's own pricing information and BLS's estimation method. Publicly available documents can be found at FTC Office of Administrative Law Judges Docket No. 9341.

consider developing hedonic techniques that work in real time for microprocessors. With more frequent and granular price quotes, the characteristics method and adjacent-period method can potentially generate reliable estimates.[28]

[28] The validity of price sources is certainly another consideration.

6 Conclusion

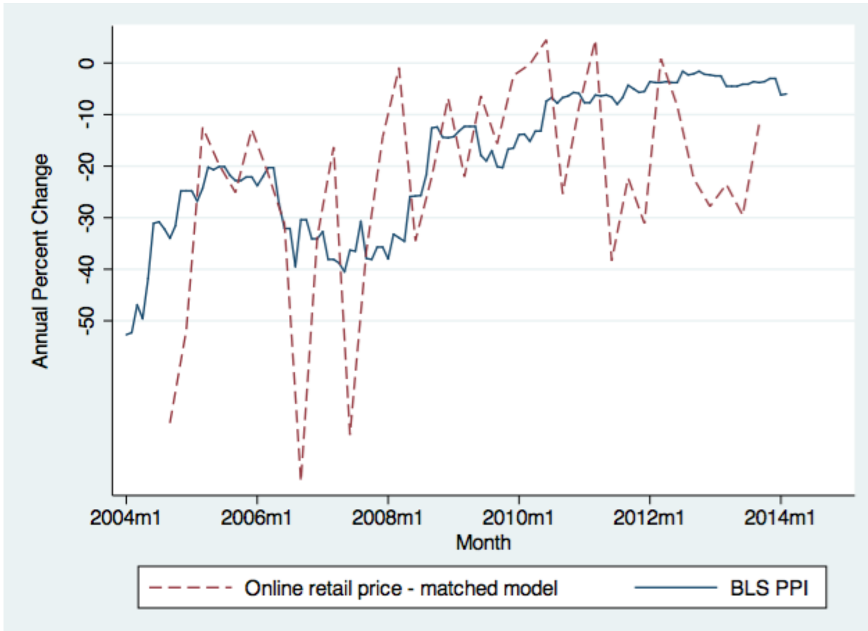
In this paper, I present estimates for quality adjusted price indices using hedonic methods. My estimates show that the laptop microprocessors are declining at -20% to -30% per year over the past ten years. I note there is a slowdown since 2010: compared with -25% to -35% per year over 2004-2010, the annual decline plateaus around -15% to -25% over 2010-2013.

At first glance, such price decline seems rather satisfactory. The quality is still improving and we will continue to benefit from advancement in IT. Nonetheless, I stress that over the previous decades, the quality adjusted price fell at -50% to -75% annually for overall microprocessors. Given the extraordinary productivity growth from IT, even the mild slowdown is a cautionary sign.

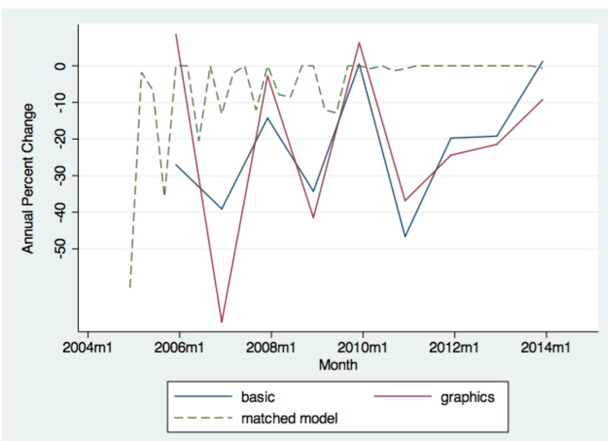
Additionally, the quality adjusted price trends provide valuable information on the industry. Even though we have all benefitted personally from the drastic decrease in quality adjusted prices, it is challenging to understand the complex technological advancement and intricate market structure fully. My analysis on the quality adjusted price trends helps decipher the progress in the microprocessor industry. With further analysis of my results, I conclude that the slower decline in quality adjust prices since 2010 are caused by both different quality improvement patterns and changing pricing strategies. Quality improvement in terms of basic performance has slowed down, while improvement in graphics performance has kept up its momentum. Moreover, energy efficiency might have achieved a gain post-2010 compared to pre-2010. The changes in pricing strategies are similarly multifaceted: increase in marginal cost and markup could have both contributed to the smaller adjustment in prices.

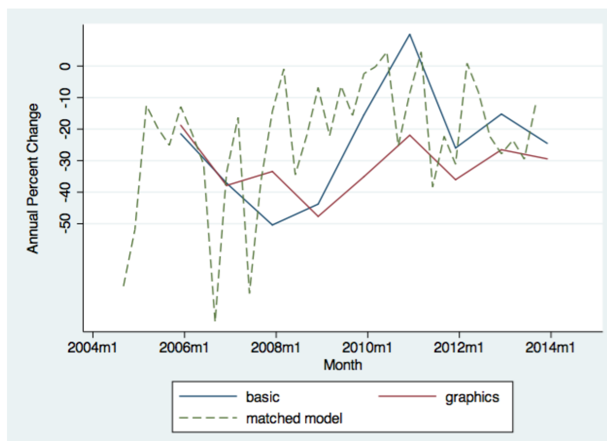
The broader implications of price trends emphasizes the importance of an accurate estimation method. Return to the question from the introduction: is the BLS PPI also biased for quality adjusted price indices for laptop microprocessors? The empirical evidence in this paper suggests that the PPI might be an underestimate of the price decline for laptop microprocessors. Since the PPI is based on a larger sample of microprocessor than just laptop microprocessors, a direct comparison between my estimates and the PPI is likely invalid. A better comparison is with the matched-model index derived from my dataset. I construct a matched-model price index from online retail prices (it is impossible to calculate matched-model index using Intel introduction prices) as shown in Figure 12 I. For most time periods, the online retail matched-model price index agree

Figure 12: Laptop Microprocessor Price Indices Comparison



I. Online Retail Price Matched Model Index





II. Intel Introduction Price Hedonic Index III. Online Retail Price Hedonic Index

Notes: The percentage change in hedonic indices are the medians of estimates using three different methods (two for characteristic method in 2011-2013 online retail prices); The matched model indices are calculated as the geometric mean of quarterly price relatives. The intel matched-model index uses quarterly Intel prices.

with the PPI. Besides the negative shocks in online retail prices, which are probably due to small sample size, the online retail matched-model price index seems to fall at a slower rate than PPI. This difference is not surprising: after all, the PPI includes a variety of microprocessors and builds on a more consistent database. Now I compare the hedonic indices with the matched-model index on from the same database.

Indeed, the estimates by the matched-model present smaller declines for 2010-2013 compared with the hedonic method as depicted in Figure 12. I demonstrate that the method employed by the PPI, mainly the matched-model method, is more vulnerable to the new goods bias than the hedonic method for the laptop microprocessor industry.

References

Ana Aizcorbe and Samuel Kortum. Moore's Law and the Semiconductor Industry: A Vintage Model. *Scandinavian Journal of Economics*, 107(4), 603-30, 2005.

Ana Aizcorbe, Carol Corrado, and Mark Doms. When Do Matched-model and Hedonic Techniques Yield Similar Price Measures? *Working Papers in Applied Economic Theory, Federal Reserve Bank of San Francisco.*, 2003.

Ana Aizcorbe, Stephen D. Oliner, and Daniel E. Sichel. Shifting Trends in Semiconductor Prices and the Pace of Technological Progress. *Business Economics*, 43(3), 23-39, 2008.

Michael R. Betker, John S. Fernand, and Shaun P. Whalen. The History of the Microprocessor. *Bell Labs Technical Journal*, Autumn, 1997.

S. Borkar and A. Chien. The Future of Microprocessors. *Communications of the ACM*, 54, 2011.

David M. Byrne, Stephen D. Oliner, and Daniel E. Sichel. How Fast Are Semiconductor Prices Falling? (Preliminary Version). 2014.

Paul Chwelos. Approaches to Performance Measurement in Hedonic Analysis: Price Indexes for Laptop Computers in the 1990Õs. *Economics of Innovation and New Technology*, 12(3), 199-224, 2003.

Ronald L. Goettler and Brett R. Gordon. Does AMD Spur Intel to Innovate More? *Journal of Political Economy*, 119, 2011.

Brett R. Gordon. A Dynamic Model of Consumer Replacement Cycles in the PC Processor Industry. *Marketing Science*, 28, 2009.

Bruce T. Grimm. Price Iindexes for Selected Ssemiconductors: 1974-96. *Survey of Current Business*, 78(2),8-24, 1998.

Gordon E. Moore. Cramming More Components onto Integrated Circuits. *Electronics*, 38(8), 1965.

Christopher Nosko. Competition and Quality Choice in the CPU Market. *unpublished manuscript*, 2010.

Ariel Pakes. A Reconsideration of Hedonic Price Indices with an Application to PC's. *American Economic Review*, 93 (5), 2003. 1578-1596.

Unni Pillai. A Model of Technological Progress in the Microprocessor Industry. *Journal of Industrial Economics*, 2013.

Jack E. Triplett. Handbook on Hedonic Indexes and Quality Adjustment in Price Indexes: Special Application to Information Technology Products. *Brookings Institution*, 2004.

List of Figures

List of Tables

Table A1: Laptop Microprocessor Price Quote Frequency

Intel Price Lists

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2004	0	1	0	1	1	0	1	1	1	1	0	1
2005	1	1	0	0	1	0	1	0	1	0	0	1
2006	1	0	0	0	0	1	1	0	0	1	0	1
2007	1	0	0	1	0	0	0	0	1	0	1	0
2008	1	0	0	1	0	0	2	3	3	1	0	2
2009	1	1	2	1	1	1	1	1	2	1	1	0
2010	0	1	0	1	0	1	1	0	0	1	1	0
2011	1	2	3	2	1	1	1	0	1	1	2	1
2012	1	1	0	1	1	1	0	0	1	1	0	1
2013	1	0	0	1	0	1	0	1	2	0	1	1

Online Retail Prices

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2004	0	0	0	1	1	1	1	1	1	1	1	1
2005	1	1	1	1	1	1	1	1	1	1	1	1
2006	1	1	1	1	1	1	1	1	1	1	1	1
2007	1	1	1	1	1	1	1	1	1	1	1	1
2008	1	1	1	1	1	1	1	1	1	1	1	1
2009	1	1	1	1	1	1	1	1	1	1	1	1
2010	1	1	1	1	1	1	1	1	1	1	1	1
2011	1	0	0	1	0	1	0	1	0	1	0	1
2012	1	0	1	0	1	0	0	1	0	0	0	1
2013	1	0	0	1	0	0	0	1	0	0	0	0

Notes: Counts of price quotes per month. For 2004q2 to 2013q3, there is at least one price quote in a quarter. So I fix the price frequency to be quarterly.

Table A2: Summary Statistics for Laptop Microprocessor Prices

Intel Initial Price				
Year	minimum	median	maximum	total count
2004	107	262	637	13
2005	134	262	637	11
2006	86	241	637	23
2007	107	262	851	19
2008	80	286.5	851	26
2009	70	316	1054	19
2010	80	272	1096	18
2011	70	317	1096	19
2012	86	346	1096	10
2013	86	225	454	12
Total	70	284	1096	170
Family	minimum	median	maximum	total count
Celeron	70	86	134	11
Celeron Dual-Core	80	83	86	2
Celeron M	86	134	161	16
Core 2 Duo	209	305	637	42
Core 2 Extreme	851	851	1038	5
Core 2 Quad	348	599.5	851	2
Core 2 Solo	241	262	262	3
Core Duo	209	294	637	9
Core Solo	209	241	262	5
Core i5	225	250	287	15
Core i7	278	378	1096	34
Legacy Celeron	80	86	134	8
Pentium M	209	284	637	18
Total	70	284	1096	170

Online Retail Price				
Year	minimum	median	maximum	total count
2004	108.68	157.39	458.63	9
2005	63.25	174.92	626.23	50
2006	49.23	118.195	1100	54
2007	28.23	98.5	1183.6	49
2008	40.9	232.57	1018	25
2009	160.36	289	990	23
2010	190	334.485	1000	26
2011	64	245	1003	67
2012	35	251	1372	122
2013	22	194.45	964.95	138
Total	22	225	1372	563
Family	minimum	median	maximum	total count
A-Series	40.99	103.875	139.99	20
Core 2 Duo	160.36	277.08	543.83	22
Core 2 Extreme	900	971.58	1183.6	19
Core 2 Quad	190	265	864	21
Core i3	22	72	125.98	26
Core i5	30	188	250	93
Core i7	40	304	1372	225
Mobile Athlon 64	82	188.16	368	27
Mobile Pentium 4	157.11	167.43	186.1	6
Mobile Sempron	28.23	73.36	135	73
Pentium 4	159.13	400	626.23	22
Pentium D	85	170.17	269.5	9
Total	22	225	1372	563

Notes: The samples include laptop microprocessors that have both price information and benchmark scores. Note that the two samples, Intel and online retail, have different microprocessor coverages. Online retail dataset includes a few desktop microprocessors that are relatively low in TDP and used to assemble laptop computers. Also, there are a few AMD microprocessors in the online retail sample.

Table A3: Laptop Microprocessor Benchmark Coverage

Intel

	SuperPI1M	wPrime32	3DMark06
2003	3	2	0
2004	19	17	9
2005			
2006	23	20	14
2007	14	14	11
2008	24	20	24
2009	18	18	16
2010	16	15	14
2011	20	19	20
2012	11	10	10
2013	9	9	9
Total	157	144	127

Online Retail

	SuperPI1M	wPrime32	3DMark06
2003	1	1	0
2004	7	7	0
2005	8	8	1
2006	4	5	4
2007	6	3	5
2008	4	4	4
2009	8	8	8
2010	10	10	9
2011	16	15	16
2012	19	19	19
2013	3	3	3
Total	86	83	69

Notes: Counts of benchmarked microprocessors by their introduction year.

Intel

	SuperPI1M	wPrime32	3DMark06
Celeron	11	10	9
Celeron Dual-Core	2	0	2
Celeron M	16	14	5
Core 2 Duo	37	39	36
Core 2 Extreme	5	5	5
Core 2 Quad	2	2	2
Core 2 Solo	2	1	2
Core Duo	9	6	6
Core Solo	5	3	1
Core i5	14	14	14
Core i7	33	32	32
Legacy Celeron	6	4	4
Pentium M	15	14	9
Total	157	144	127

Online Retail

	SuperPI1M	wPrime32	3DMark06
A-Series	4	4	4
Core 2 Duo	3	3	3
Core 2 Extreme	3	1	3
Core 2 Quad	2	2	2
Core i3	5	5	5
Core i5	13	13	13
Core i7	36	35	35
Mobile Athlon 64	6	6	0
Mobile Pentium 4	1	1	0
Mobile Sempron	9	9	3
Pentium 4	3	3	0
Pentium D	1	1	1
Total	86	83	69

Notes: Counts of benchmarked microprocessors by microprocessor family.

Table A4: Dummy Variable Method Index Details

Intel Introduction Price Basic Performance									
Quarter	δ	Annual LD	Annual Index	Annual Growth	Quarter	δ	Annual LD	Annual Index	Annual Growth
2004q3					2009q1	-1.742***			
						(0.362)			
2004q4					2009q2	-1.268***			
						(0.224)			
2005q1	-0.119				2009q3	-0.999***	0.445*	262.479	56.101
	(0.175)					(0.217)	(0.277)		
2005q2					2009q4				
2005q3	-0.565		311.491		2010q1	-1.921***			
	(0.359)					(0.246)			
2005q4					2010q2	-1.309***			
						(0.252)			
2006q1	-0.463**				2010q3	-1.964***	-0.965***	100.000	-61.902
	(0.201)					(0.240)	(0.193)		
2006q2	-0.756***				2010q4				
	(0.176)								
2006q3					2011q1	-2.052***			
						(0.246)			
2006q4	-0.716***	-0.12	276.047	-11.379	2011q2	-2.273***			
	(0.185)	(0..294)				(0.312)			
2007q1	-0.638*				2011q3	-2.022***			
	(0.360)					(0.303)			
2007q2					2011q4	-1.919***	0.036	103.666	3.666
						(0.274)	(0.239)		
2007q3	-0.720***	-0.005	274.578	-0.532	2012q1	-2.091***			
	(0.179)	(0.223)				(0.367)			
2007q4					2012q2	-2.260***			
						(0.283)			
2008q1	-0.915***				2012q3	-2.580***	-0.88*	42.941	-58.577
	(0.196)					(0.431)	(0.608)		
2008q2					2012q4				
2008q3	-1.199***				2013q1	-2.351***			
	(0.194)					(0.284)			
2008q4	-1.333***	-0.49***	168.147	-38.762	2013q2	-2.267***			
	(0.245)	(0.166)				(0.353)			
					2013q3	-2.384***			
						(0.360)			
					2013q4	-2.153***	0.34*	60.427	40.720
						(0.233)	(0.323)		

<u>*Notes:*</u> *** $p < 0.01$, ** $p < 0.05$, * $p < 0.1$.
The significance test I focus on is whether the annual log difference is significantly different from zero in the "Annual LD" columns.

Intel Introduction Price
Graphics Performance

Quarter	δ	Annual LD	Annual Index	Annual Growth	Quarter	δ	Annual LD	Annual Index	Annual Growth
2004q3					2009q1	-1.919***			
						(0.378)			
2004q4	0.516		650.215		2009q2	-1.539***			
	(0.456)					(0.338)			
2005q1	0.267				2009q3	-1.000***	0.805***	286.910	123.744
	(0.312)					(0.330)	(0.269)		
2005q2					2009q4				
2005q3	0.578	0.083***	706.251	8.618	2010q1	-1.753***			
	(0.456)	(0.324)				(0.347)			
2005q4					2010q2	-1.235***			
						(0.356)			
2006q1	-0.674**				2010q3	-2.054***	-1.054***	100.000	-65.146
	(0.334)					(0.354)	(0.213)		
2006q2					2010q4				
2006q3	-0.709*				2011q1	-2.130***			
	(0.393)					(0.360)			
2006q4	-0.927***	-1.204	211.869	-70.001	2011q2	-2.302***			
	(0.315)	(0.252)				(0.419)			
2007q1	-0.691				2011q3	-2.099***			
	(0.487)					(0.411)			
2007q2					2011q4	-1.711***	0.27*	131.574	31.574
						(0.374)	(0.274)		
2007q3	-0.853***	0.098	233.840	10.370	2012q1	-2.137***			
	(0.302)	(0.252)				(0.464)			
2007q4					2012q2	-2.356***			
						(0.396)			
2008q1	-1.119***				2012q3	-2.740***	-1.372***	33.367	-74.640
	(0.330)					(0.537)	(0.695)		
2008q2					2012q4				
2008q3	-1.314***				2013q1	-2.510***			
	(0.320)					(0.397)			
2008q4	-1.604***	-0.61***	128.231	-45.163	2013q2	-2.417***			
	(0.329)	(0.152)				(0.464)			
					2013q3	-2.471***			
						(0.450)			
					2013q4	-2.370***	0.296	44.861	34.447
						(0.353)	(0.354)		

Online Retail Price
Basic Performance

Quarter	δ	Annual LD	Annual Index	Annual Growth	Quarter	δ	Annual LD	Annual Index	Annual Growth
2004q3	0.026				2009q1	-2.109***			
	(0.606)					(0.583)			
2004q4	0.107		1004.425		2009q2	-2.210***			
	(0.575)					(0.597)			
2005q1	0.231				2009q3	-2.086***			
	(0.557)					(0.579)			
2005q2	0.027				2009q4	-2.093***	-0.168	111.293	-15.465
	(0.546)					(0.579)	(0.304)		
2005q3	-0.070				2010q1	-2.053***			
	(0.541)					(0.573)			
2005q4	-0.084	-0.191	829.788	-17.387	2010q2	-2.064***			
	(0.543)	(0.272)				(0.573)			
2006q1	-0.161				2010q3	-2.162***			
	(0.546)					(0.579)			
2006q2	-0.287				2010q4	-2.200***	-0.107	100	-10.147
	(0.543)					(0.579)	(0.302)		
2006q3	-0.498				2011q1	-2.487***			
	(0.539)					(0.580)			
2006q4	-0.844	-0.76***	388.064	-53.233	2011q2	-2.332***			
	(0.548)	(0.212)				(0.550)			
2007q1	-0.845				2011q3	-2.383***			
	(0.548)					(0.550)			
2007q2	-1.085**				2011q4	-2.418***	-0.218	80.413	-19.587
	(0.548)					(0.548)	(0.24)		
2007q3	-1.284**				2012q1	-2.345***			
	(0.551)					(0.548)			
2007q4	-1.197**	-0.353*	272.645	-29.742	2012q2	-2.457***			
	(0.557)	(0.244)				(0.548)			
2008q1	-1.136**				2012q3	-2.510***			
	(0.562)					(0.548)			
2008q2	-1.501**				2012q4	-2.584***	-0.166	68.113	-15.295
	(0.588)					(0.543)	(0.139)		
2008q3	-1.784***				2013q1	-2.618***			
	(0.589)					(0.543)			
2008q4	-1.925***	-0.728***	131.653	-51.713	2013q2	-2.754***			
	(0.572)	(0.285)				(0.543)			
					2013q3	-2.784***	-0.267***	52.17	-23.407
						(0.543)	(0.145)		

Online Retail Price
Graphics Performance

Quarter	δ	Annual LD	Annual Index	Annual Growth	Quarter	δ	Annual LD	Annual Index	Annual Growth
2004q3					2009q1	-1.686***			
						(0.558)			
2004q4					2009q2	-1.666***			
						(0.566)			
2005q1					2009q3	-1.790***			
						(0.559)			
2005q2					2009q4	-1.796***	-0.167	110.296	-15.38
						(0.559)	(0.325)		
2005q3	0.173		764.46		2010q1	-1.757***			
	(0.625)					(0.554)			
2005q4	0.140				2010q2	-1.768***			
	(0.625)					(0.554)			
2006q1	0.039				2010q3	-1.856***			
	(0.625)					(0.560)			
2006q2	-0.304				2010q4	-1.894***	-0.098	100	-9.335
	(0.572)					(0.560)	(0.322)		
2006q3	-0.493				2011q1	-2.276***			
	(0.547)					(0.561)			
2006q4	-0.561	-0.701	379.24	-50.391	2011q2	-2.213***			
	(0.547)	(0.417)				(0.535)			
2007q1	-0.552				2011q3	-2.263***			
	(0.539)					(0.535)			
2007q2	-0.831				2011q4	-2.352***	-0.458***	63.255	-36.745
	(0.539)					(0.534)	(0.165)		
2007q3	-0.969*				2012q1	-2.296***			
	(0.537)					(0.533)			
2007q4	-0.968*	-0.407*	252.439	-33.436	2012q2	-2.414***			
	(0.540)	(0.261)				(0.533)			
2008q1	-1.051*				2012q3	-2.467***			
	(0.549)					(0.533)			
2008q2	-1.357**				2012q4	-2.698***	-0.346***	44.754	-29.249
	(0.575)					(0.531)	(0.146)		
2008q3	-1.575***				2013q1	-2.731***			
	(0.576)					(0.531)			
2008q4	-1.629***	-0.661***	130.343	-48.367	2013q2	-2.850***			
	(0.558)	(0.306)				(0.531)			
					2013q3	-2.912***	-0.285**	33.644	-24.824
						(0.530)	(0.139)		

Table A5: Dummy Variable Method Index

Annual Growth

	I		II	
Year	(a)	(b)	(a)	(b)
2005		8.618†	-17.387	
2006	-11.379 †	-70.001†	-53.233	-50.391†
2007	-0.532 †	10.37	-29.742	-33.436
2008	-38.762	-45.163	-51.713	-48.367
2009	56.101	123.744	-15.465 †	-15.38 †
2010	-61.902	-65.146	-10.147 †	-9.33†5
2011	3.666 †	31.574	-19.587 †	-36.745
2012	-58.577	-74.64	-15.295 †	-29.249
2013	40.72	34.447 †	-23.407	-24.824

Notes: The † indicates a change insignificantly from zero at 10% level.

Summary of CAGR

	I		II	
Year	(a)	(b)	(a)	(b)
2004-2010	-20.327	-26.803	-31.921	-33.422
2010-2013	-15.457***	-23.448***	-19.498***	-30.449***
Overall	-18.535	-25.702	-28.009	-32.323

Notes: Price sources: Panel I: Intel introduction prices; Panel II: online retail prices. Benchmark variables: Column (a): basic (gmean of SuperPI1M and wPrime32); Column (b): graphics (3DMark06). The start year depends on the availability of data to generate index estimates.
*** $p < 0.01$, ** $p < 0.05$, * $p < 0.1$.
The significance test is whether CAGR in 2010-2013 differs from that in 2004-2010.

Table A6: Characteristics Method

Intel Introduction Price

Year	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
ln(basic)	-2.181**	-3.527***	-2.082***	-1.929***	-0.602*	-2.051***	-1.134**	-1.489***	-1.424***	-1.217***
	(0.579)	(0.618)	(0.196)	(0.365)	(0.336)	(0.426)	(0.442)	(0.155)	(0.263)	(0.223)
ln(TDP)	-0.456	-0.815***	-0.821***	-0.483	-0.015	-0.773**	-0.158	-0.132	-0.119	-0.401
	(0.471)	(0.176)	(0.142)	(0.337)	(0.245)	(0.348)	(0.523)	(0.179)	(0.290)	(0.294)
Observations	8	8	19	11	22	15	15	18	9	11
R-squared	0.637	0.814	0.862	0.748	0.224	0.722	0.448	0.860	0.815	0.739

Intel Introduction Price

Year	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
ln(graphics)	3.934*	2.711	2.958***	1.699***	1.731***	2.434***	1.469**	1.327***	1.316***	1.218***
	(0.471)	(1.549)	(0.317)	(0.456)	(0.431)	(0.463)	(0.556)	(0.151)	(0.321)	(0.296)
ln(TDP)			-1.195***	-0.720**	-0.982***	-1.433***	-0.459	-0.290	-0.148	-0.674
			(0.209)	(0.315)	(0.316)	(0.446)	(0.603)	(0.191)	(0.370)	(0.393)
Observations	3	6	12	12	24	16	15	19	9	11
R-squared	0.972	0.292	0.888	0.564	0.402	0.695	0.460	0.825	0.713	0.612

Online Retail Price

Year	2004	2005	2006	2007	2008	2009	2010
ln(basic)	-2.392**	-2.328***	-1.799***	-1.841***	-1.217***	0.404	4.177***
	(0.852)	(0.293)	(0.202)	(0.177)	(0.258)	(0.723)	(1.343)
ln(TDP)	0.582*	0.740***	0.487***	0.053	-0.301	2.542**	6.329***
	(0.254)	(0.104)	(0.137)	(0.214)	(0.487)	(0.936)	(1.293)
Observations	9	50	54	40	21	21	26
R-squared	0.533	0.787	0.697	0.817	0.676	0.479	0.597

Online Retail Price

Year	2004	2005	2006	2007	2008	2009	2010
ln(graphics)		0.881***	1.618***	1.689***	-0.266	1.385	5.852***
		(0.068)	(0.188)	(0.200)	(1.132)	(0.905)	(0.607)
ln(TDP)			-0.353*	-0.156	1.744	1.272	-1.167**
			(0.195)	(0.254)	(1.210)	(0.826)	(0.465)
Observations		5	20	37	21	23	26
R-squared		0.977	0.826	0.843	0.497	0.534	0.886

Notes: *** $p < 0.01$, ** $p < 0.05$, * $p < 0.1$.

Table A7: Characteristics Method Index Details

Intel Introduction Price
Basic Performance

Year	$h^{t+1}_{char}(\hat{x}_{i,t+1})$	$h^{t+2}_{char}(\hat{x}_{i,t+1})$	Laspeyres	$h^{t+1}_{char}(\hat{x}_{i,t+2})$	$h^{t+2}_{char}(\hat{x}_{i,t+2})$	Paasche	Fisher
2004	259.158			281.429			
2005	231.918	202.476	0.781	760.651	231.918	0.824	0.802
2006	228.771	126.584	0.546	360.615	228.771	0.301	0.405
2007	312.791	193.881	0.847	493.206	312.791	0.867	0.857
2008	304.313	273.589	0.875	346.046	304.313	0.617	0.735
2009	378.954	280.601	0.922	578.282	378.954	1.095	1.005
2010	283.116	220.2	0.581	365.329	283.116	0.490	0.533
2011	296.633	211.906	0.748	419.131	296.633	0.812	0.780
2012	336.155	241.4	0.814	165.891	336.155	0.802	0.808
2013		297.534	0.885		189.301	1.141	1.005

Intel Introduction Price
Graphics Performance

Year	$h^{t+1}_{char}(\hat{x}_{i,t+1})$	$h^{t+2}_{char}(\hat{x}_{i,t+1})$	Laspeyres	$h^{t+1}_{char}(\hat{x}_{i,t+2})$	$h^{t+2}_{char}(\hat{x}_{i,t+2})$	Paasche	Fisher
2004	356.042			427.653			
2005	369.421	325.587	0.914	2206.428	369.421	0.864	0.889
2006	327.106	50.153	0.136	336.484	327.106	0.148	0.142
2007	325.918	319.02	0.975	469.267	325.918	0.969	0.972
2008	279.867	189.407	0.581	300.173	279.867	0.596	0.589
2009	321.093	295.618	1.056	485.65	321.093	1.070	1.063
2010	283.116	219.731	0.684	424.143	283.116	0.583	0.632
2011	297.671	206.972	0.731	425.566	297.671	0.702	0.716
2012	336.155	232.516	0.781	177.734	336.155	0.790	0.785
2013		265.94	0.791		185.158	1.042	0.908

Online Retail Price
Basic Performance

Year	$h^{t+1}_{char}(\hat{x}_{i,t+1})$	$h^{t+2}_{char}(\hat{x}_{i,t+1})$	Laspeyres	$h^{t+1}_{char}(\hat{x}_{i,t+2})$	$h^{t+2}_{char}(\hat{x}_{i,t+2})$	Paasche	Fisher
2004	162.92			219.605			
2005	173.515	127.137	0.780	215.956	173.515	0.790	0.785
2006	141.364	118.309	0.682	263.637	141.364	0.655	0.668
2007	132.571	69.154	0.489	328.293	132.571	0.503	0.496
2008	174.466	105.132	0.793	407.729	174.466	0.531	0.649
2009	338.339	167.097	0.958	366.884	338.339	0.830	0.891
2010		426.881	1.262		390.489	1.064	1.159

Online Retail Price
Graphics Performance

Year	$h^{t+1}_{char}(\hat{x}_{i,t+1})$	$h^{t+2}_{char}(\hat{x}_{i,t+1})$	Laspeyres	$h^{t+1}_{char}(\hat{x}_{i,t+2})$	$h^{t+2}_{char}(\hat{x}_{i,t+2})$	Paasche	Fisher
2004	193.563			211.293			
2005	173.423	127.534	0.659	340.414	173.423	0.821	0.735
2006	239.05	112.791	0.650	563.801	239.05	0.702	0.676
2007	317.065	227.137	0.950	465.141	317.065	0.562	0.731
2008	368.382	155.915	0.492	432.141	368.382	0.792	0.624
2009	338.339	248.494	0.675	367.482	390.489	0.904	0.781
2010		442.851	1.309		390.489	1.063	1.179

Notes: Fisher (chain weighted) index is the geometric mean of Laspeyres (base-period weighted) index and Paasche (current-period weighted) index.

Table A8: Characteristics Method - Fisher Index

Annual Growth

	I		II	
Year	(a)	(b)	(a)	(b)
2005	-19.761	-11.121	-21.477	
2006	-59.484	-85.813	-33.192	-26.462
2007	-14.262	-2.807	-50.402	-32.419
2008	-26.537	-41.128	-35.082	-26.901
2009	0.487	6.297	-10.851	-37.594
2010	-46.663	-36.839	15.882	-21.927
2011	-22.043	-28.371		
2012	-19.211	-21.45		
2013	0.499	-9.216		

Summary of CAGR

	I		II	
Year	(a)	(b)	(a)	(b)
2004-2010	-30.806	-39.624	-25.247	-29.271
2010-2013	-14.14	-20.063	n/a	n/a
Overall	-25.646	-33.703	n/a	n/a

Notes: Price sources: Panel I: Intel introduction prices; Panel II: online retail prices. Benchmark variables: Column (a): basic (gmean of SuperPI1M and wPrime32); Column (b): graphics (3DMark06).

Table A9: Adjacent-Year Method

Intel Introduction Price

Year	2004-2005	2005-2006	2006-2007	2007-2008	2008-2009	2009-2010	2010-2011	2011-2012	2012-2013
ln(basic)	-2.403***	-2.171***	-2.130***	-1.402***	-1.297***	-1.444***	-1.316***	-1.447***	-1.299***
	(0.381)	(0.170)	(0.177)	(0.306)	(0.303)	(0.327)	(0.162)	(0.126)	(0.169)
Quarter	-0.105*	-0.124***	-0.053**	-0.105**	-0.006	-0.133***	-0.055*	-0.047*	0.003
	(0.049)	(0.017)	(0.022)	(0.046)	(0.035)	(0.047)	(0.028)	(0.026)	(0.039)
Observations	16	27	30	33	37	30	33	27	20
R-squared	0.729	0.861	0.847	0.449	0.502	0.551	0.720	0.856	0.803
ln(graphics)	2.811	1.872***	2.054***	1.675***	2.148***	2.075***	1.274***	1.294***	1.227***
	(1.914)	(0.479)	(0.304)	(0.280)	(0.300)	(0.376)	(0.152)	(0.132)	(0.219)
Quarter	0.061	-0.207***	-0.026	-0.134***	0.003	-0.100*	-0.070**	-0.048	-0.026
	(0.307)	(0.056)	(0.029)	(0.033)	(0.035)	(0.049)	(0.028)	(0.029)	(0.044)
Observations	9	18	24	36	40	31	34	28	20
R-squared	0.343	0.440	0.672	0.538	0.597	0.579	0.720	0.802	0.702

Online Retail Price

Year	2004-2005	2005-2006	2006-2007	2007-2008	2008-2009	2009-2010	2010-2011	2011-2012	2012-2013
ln(basic)	-2.412***	-2.012***	-1.894***	-1.584***	-1.086***	1.229*	-1.475***	-1.768***	-1.770***
	(0.258)	(0.150)	(0.118)	(0.144)	(0.224)	(0.643)	(0.304)	(0.146)	(0.111)
Quarter	-0.074***	-0.116***	-0.189***	-0.144***	-0.081*	0.024	-0.098***	-0.041**	-0.074***
	(0.023)	(0.017)	(0.023)	(0.032)	(0.042)	(0.028)	(0.024)	(0.017)	(0.017)
Observations	59	104	94	61	42	47	87	176	250
R-squared	0.788	0.762	0.807	0.779	0.664	0.504	0.390	0.479	0.541
ln(graphics)	0.846***	1.648***	1.790***	1.716***	0.694	3.155***	1.465***	1.661***	1.843***
	(0.046)	(0.170)	(0.133)	(0.198)	(0.751)	(0.644)	(0.298)	(0.147)	(0.113)
Quarter	-0.052	-0.119**	-0.131***	-0.162***	-0.108**	-0.066***	-0.109***	-0.068***	-0.104***
	(0.023)	(0.043)	(0.029)	(0.032)	(0.043)	(0.024)	(0.024)	(0.018)	(0.018)
Observations	5	25	57	58	44	49	87	177	254
R-squared	0.990	0.834	0.862	0.805	0.523	0.653	0.394	0.440	0.537

Notes: *** $p < 0.01$, ** $p < 0.05$, * $p < 0.1$.

Table A10: Adjacent Year Method Index

Annual Growth

	I		II	
Year	(a)	(b)	(a)	(b)
2005	-34.295	27.634	-25.621	-18.779†
2006	-39.104	-56.308	-37.124	-37.874
2007	-19.104	-9.877	-53.046	-40.785
2008	-34.295	-41.492	-43.786	-47.691
2009	-2.371†	1.207†	-27.675	-35.079
2010	-41.257	-32.968	10.076†	-23.203
2011	-19.748	-24.422	-32.43	-35.338
2012	-17.139	-17.469†	-15.126	-23.815
2013	1.207†	-9.877†	-25.621	-34.032

Notes:
The † indicates a change insignificantly from zero at 10% level.

Summary of CAGR

	I		II	
Year	(a)	(b)	(a)	(b)
2004-2010	-29.578	-23.56	-32.068	-34.645
2010-2013	-12.366	-17.469	-24.723	-31.248
Overall	-24.253	-21.582	-29.703	-33.532

Notes: Price sources: Panel I: Intel introduction prices; Panel II: online retail prices. Benchmark variables: Column (a): basic (gmean of SuperPI1M and wPrime32); Column (b): graphics (3DMark06). The start year depends on the availability of data to generate index estimates.

Appendix 1: Related Proofs for Hedonic Functions

This appendix includes proofs on the conditions under which the matched-model and the hedonic method yield similar price index estimates.

In general, when there is no turnover in the market and the quality features remain constant over adjacent periods, the two methods would yield similar estimates.

<u>DV Method</u>

Denote N_t the number of goods that have price information at time period t. The DV method index number formula is

$$index\{\frac{t}{t-1}\} = exp(\delta_t - \delta_{t-1}) = \frac{\dfrac{\prod\limits_{n \in N_t} (P_{n,t})^{1/N_t}}{\prod\limits_{n \in N_{t-1}} (P_{n,t-1})^{1/N_{t-1}}}}{\dfrac{\prod_{n \in N_t} (x_n)^{\beta/N_t}}{\prod_{n \in N_{t-1}} (x_n)^{\beta/N_{t-1}}}}$$

<u>**Claim:**</u> When $N_{t-1} = N_t = N$ and x_i are fixed quality features, the index is $\prod\limits_{n \in N} (\frac{P_{n,t}}{P_{n,t-1}})^{1/N}$, the matched-model geometric mean index. That is, the DV method and the matched-model method yield the same estimates.

<u>*Proof:*</u> The denominator of Equation 2, the hedonic quality adjustment, is given by $\frac{\prod_{n \in N_t} (x_n)^{\beta/N_t}}{\prod_{n \in N_{t-1}} (x_n)^{\beta/N_{t-1}}}$. When there is no turnover, the number of goods stays constant from $t-1$ to t. So $N_{t-1} = N_t = N$. Additionally since x_n is a fixed quality feature, the quality adjustment is equal to one.

Now Equation 2 becomes

$$index\{\frac{t}{t-1}\} = \frac{\prod\limits_{n \in N} (P_{n,t})^{1/N}}{\prod\limits_{n \in N} (P_{n,t-1})^{1/N}}$$

which is $\prod\limits_{n \in N} (\frac{P_{n,t}}{P_{n,t-1}})^{1/N}$. Then the DV index is the geometric mean of price relatives.

This proof follows similarly from Aizcorbe et al. [2003]. Aizcorbe et al. [2003] also confirm with empirical evidence. In practice, if the price quotes are frequent enough, then the turnover rate between two adjacent time periods is likely low. With query set up to retrieve prices from online price search engines at desired frequency, it might be feasible to establish more granular price sets. □

Characteristics Method

To construct characteristics price index, we predict prices using characteristics functions from two overlapping time periods $t+1$ and $t+2$:

$$\begin{aligned} h_{char}^{t+1} : ln(P_{i,t+1}) &= \alpha_{t+1} + \beta_{t+1} ln(x_{i,t+1}) + \epsilon_{i,t+1} \\ h_{char}^{t+2} : ln(P_{i,t+2}) &= \alpha_{t+2} + \beta_{t+2} ln(x_{i,t+2}) + \epsilon_{i,t+2}. \end{aligned}$$

Claim: When $N_{t-1} = N_t = N$ and x_i are fixed quality features, the three superlative index formulae, the Laspeyres, Paasche and Fisher, yield the same index. Moreover, the characteristics method and the matched-model method yield the same estimates.

Proof: When x_i are fixed quality features, we have $x_{i,t+1} = x_{i,t+2} = x_i$. Then the Laspeyres index is $\frac{h_{char}^{t+2}(x_i)}{h_{char}^{t+1}(x_i)}$. The Paasche index is $\frac{h_{char}^{t+2}(x_i)}{h_{char}^{t+1}(x_i)}$. These two are the same and their geometric average, the Fisher index, is therefore identical as well.

In particular, since I apply the index on the average good, I take the geometric mean of the predicted price. The Laspeyres index is therefore $\frac{exp(\overline{ln(h_{char}^{t+2}(x_i))}}{exp(\overline{ln(h_{char}^{t+1}(x_i))}}$. That is,

$$\begin{aligned} \frac{exp(\overline{ln(h_{char}^{t+2}(x_i))}}{exp(\overline{ln(h_{char}^{t+1}(x_i))}} &= \frac{exp(\overline{ln(\hat{P_{i,t+2}}))}}{exp(\overline{ln(\hat{P_{i,t+1}}))}} \\ &= \frac{exp(\overline{\hat{\alpha_{t+2}} + \hat{\beta_{t+2}} ln(x_i))}}{exp(\overline{\hat{\alpha_{t+1}} + \hat{\beta_{t+1}} ln(x_i))}} \\ &= \frac{exp(\hat{\alpha_{t+2}} + \hat{\beta_{t+2}} \overline{ln(x_i)})}{exp(\hat{\alpha_{t+1}} + \hat{\beta_{t+1}} \overline{ln(x_i)})}. \end{aligned}$$

Note that by the assumption of OLS, $\hat{\alpha_{t+1}} = \overline{ln(P_{i,t+1})} - \hat{\beta_{t+1}} \overline{ln(x_i)}$. Substitute $\hat{\alpha_{t+1}}$ and $\hat{\alpha_{t+2}}$ back in the fraction, we get

$$
\begin{aligned}
&= \frac{exp(\overline{ln(P_{i,t+2})} - \hat{\beta_{t+2}}\overline{ln(x_i)} + \hat{\beta_{t+2}}\overline{ln(x_i)})}{exp(\overline{ln(P_{i,t+1})} - \hat{\beta_{t+1}}\overline{ln(x_i)} + \hat{\beta_{t+1}}\overline{ln(x_i)})} \\
&= \frac{exp(\overline{ln(P_{i,t+2})})}{exp(\overline{ln(P_{i,t+1})})}
\end{aligned}
$$

Note that $\frac{exp(\overline{ln(P_{i,t+2})})}{exp(\overline{ln(P_{i,t+1})})}$ equals $\prod_{i \in N}(\frac{P_{i,t+2}}{P_{i,t+1}})^{1/N}$ since the sample of goods is constant. Then the characteristics index is the geometric mean of price relatives. That is, the characteristics index would yield same results as the matched-model index. □

Appendix 2: Additional Details on Data

This appendix describes in additional detail the data sets used in my analysis.

Intel Official Prices

I collected the official price lists from 2004 and onward at the Intel investor relations website. The 2004-2006 price lists are available roughly every quarter on the internet archive. In 2005, the laptop microprocessors are marketed as part of Centrino chipsets, which include motherboards and a specific type of wireless network card. I exclude these products from the laptop microprocessor market. The 2006-2013 price lists are generously provided by Dr. David Byrne.

There is little documentation on the price lists. The prices are wholesale prices, for microprocessors sold in units of 1,000. There is no indication on when the price adjustment takes place or whether there should be any adjustment on prices. It is worth clarifying that not all Intel laptop microprocessors are included in the price lists. It is possible that the sales for these microprocessors are less than 1,000, but this speculation is not verified.

Online Retail Market

The other source of prices is the online retail market. The 2004-2011 prices are available weekly from SharkyExtreme, which uses PriceWatch.[29] The 2011-2013 prices are collected directly PriceWatch, a price search engine that monitors prices from Amazon, eBay and Newegg for price quotes and tracks the lowest price. I rely on the "wayback machine", a website archive, to retrieve historic price information. The frequency of these price quotes is limited by the times that PriceWatch has a "snapshot" on the "wayback machine". On average, there is at least one "snapshot" of PriceWatch in every quarter 2011-2013.

Although I call this source the online retail prices, most of the prices listed are for the "OEM" versions of the microprocessors, also referred to as "tray" or non-retail. The OEM microprocessors usually do not come equipped with a heat sink or fan. The actual retail prices are higher since they generally include the longer warranties, heat sinks and fan which OEM prices do not. Therefore these prices are comparable to those on the Intel website. There might be a few actual

[29] Available at `http://www.sharkyextreme.com/guides/WCPG/archives/`. I am very grateful to Dr. David Byrne for sharing the prices he collected.

retail prices, and I assume the additional charge is small.

Note that the retail prices do not include any additional charges for shipping, sales tax, etc.

Table A11 and Table A12 summarize the price information and quality features for each microprocessor in the analysis.

Table A11: Summary for Intel Introduction Price Dataset

Model	Introduction	Intro Price	SuperPI1M	wPrime32	3DMark06	TDP(Watt)
Intel Celeron M 340	2003	134	90	125		21
Intel Celeron M 320	2003	107	68			21
Intel Pentium M 718	2003	284		139		10
Intel Pentium M 713	2003	241	79			5
Intel Celeron M 360	2004	134	76	131		21
Intel Celeron M 370	2004	134	60	121		21
Intel Celeron M 380	2004	134	56	111		21
Intel Celeron M 373	2004	161	154			5
Intel Celeron M 390	2004	107	59	102		21
Intel Pentium M 715	2004	209	50	115		21
Intel Pentium M 735	2004	294	45	104	698	21
Intel Pentium M 760	2004	423	42	86	830	27
Intel Pentium M 740	2004	241	47	102	737	27
Intel Pentium M 780	2004	637		80	933	27
Intel Pentium M 765	2004	637	40		864	21
Intel Pentium M 755	2004	637	40	89		21
Intel Pentium M 730	2004	209	48	108	663	27
Intel Pentium M 725	2004	241	35	111	684	21
Intel Pentium M 770	2004	637			740	27
Intel Pentium M 733	2004	262	64			5
Intel Pentium M 738	2004	284	55	128		10
Intel Pentium M 745	2004	423	46	103		21
Intel Pentium M 753	2004	262	63	146		6
Intel Pentium M 758	2004	284	52	117		10
Intel Pentium M 750	2004	294	39	95	803	27
Intel Celeron M 420	2006	107	44	109		27
Intel Celeron M 440	2006	107	39	93	980	27
Intel Celeron M 430	2006	134	41	101		27
Intel Celeron M 410	2006	86	46	119		27
Intel Core 2 Duo T7400	2006	423	24	43	1862	34
Intel Core 2 Duo U7600	2006	289	43	69	1008	10
Intel Core 2 Duo T5600	2006	241	31	49	1560	34
Intel Core 2 Duo U7500	2006	262			875	10
Intel Core 2 Duo T7200	2006	294	26	46	1718	34
Intel Core 2 Duo T7600	2006	637	22	38	2005	34
Intel Core 2 Duo U7700	2006	289	46	74	1047	10
Intel Core Duo U2500	2006	289	49		922	9
Intel Core Duo T2600	2006	637	28	40	1766	31
Intel Core Duo T2300E	2006	209	35	52		31
Intel Core Duo L2500	2006	316	33			15
Intel Core Duo U2400	2006	262	55	82		9
Intel Core Duo T2400	2006	294	32	48	1488	31
Intel Core Duo T2500	2006	423	30	44	1621	31
Intel Core Duo T2700	2006	637	25		1860	31
Intel Core Duo T2300	2006	241	35	52	1380	31
Intel Core Solo U1400	2006	262	49	135		6
Intel Core Solo T1400	2006	209	33	95		27
Intel Core Solo T1300	2006	209	37	105		27
Intel Core Solo U1300	2006	241	55			5
Intel Celeron 550	2006	134		89		31
Intel Celeron M 530	2007	107	38	105	739	30
Intel Celeron M 520	2007	134	39	115		30
Intel Core 2 Duo T5500	2007	209	33	55	1392	34
Intel Core 2 Duo T7250	2007	209	30	45	1705	35
Intel Core 2 Duo T7800	2007	530	19	35	2258	35
Intel Core 2 Duo L7700	2007	316	34			17
Intel Core 2 Duo T7300	2007	241	24	46	1729	34
Intel Core 2 Duo T7700	2007	530	21	42	2058	34
Intel Core 2 Duo L7300	2007	284		65		17
Intel Core 2 Duo T7500	2007	316	23	38	1907	35
Intel Core 2 Duo T7100	2007	209	31	50	1536	34
Intel Core 2 Duo L7500	2007	316		62		17
Intel Core 2 Extreme X7900	2007	851	19	31	2449	44
Intel Core 2 Extreme X7800	2007	851	20	32	2079	44
Intel Core 2 Solo U2100	2007	241	58			6
Intel Core Solo U1500	2007	262	49		545	5
Intel Celeron 540	2007	134		110		30
Intel Celeron Dual Core T1600	2008	80	36		1350	35
Intel Celeron M 723	2008	161	44	129	555	5
Intel Core 2 Duo P8600	2008	241	21	35	2155	25
Intel Core 2 Duo T9600	2008	530	16	28	2517	35
Intel Core 2 Duo T9500	2008	530	18	66	2352	35
Intel Core 2 Duo T8100	2008	209	23	38	1878	35
Intel Core 2 Duo SL9300	2008	284	39	112		17
Intel Core 2 Duo SU9300	2008	262			1028	10
Intel Core 2 Duo P9500	2008	348	18	31	2311	25
Intel Core 2 Duo SU9400	2008	289	33	59	1197	10
Intel Core 2 Duo T9400	2008	316	18	31	2304	35
Intel Core 2 Duo T8300	2008	241	21	38	2143	35
Intel Core 2 Duo T9300	2008	316	18	32	2258	35
Intel Core 2 Duo P8400	2008	209	22	34	2036	25
Intel Core 2 Duo SP9400	2008	316	18	38	2059	25
Intel Core 2 Duo SP9300	2008	284	21	39	1974	25
Intel Core 2 Duo SL9400	2008	316	23	43	1300	17
Intel Core 2 Extreme X9100	2008	851	13	26	2810	44
Intel Core 2 Extreme QX9300	2008	1038	18	16	3780	45
Intel Core 2 Extreme X9000	2008	851	17	24	2549	44
Intel Core 2 Quad Q9100	2008	851	20	19	3310	45
Intel Core 2 Solo SU3300	2008	262			564	6
Intel Celeron T1700	2008	86	34		1456	35
Intel Celeron T1600	2008	80	36		1350	35
Intel Celeron 560	2008	134	32	79		31
Intel Celeron T3100	2008	86	30		1687	35

Model	Introduction	Intro Price	SuperPI1M	wPrime32	3DMark06	TDP(Watt)
Intel Celeron 900	2009	70	28	80		35
Intel Celeron Dual Core T1700	2009	86	34		1456	35
Intel Celeron M 743	2009	107	42	122	582	10
Intel Core 2 Duo P9600	2009	348	19	35	2189	25
Intel Core 2 Duo P9700	2009	348	16	37	2540	28
Intel Core 2 Duo T9550	2009	316	17	29	2385	35
Intel Core 2 Duo P8800	2009	241	19	31	2355	25
Intel Core 2 Duo SP9600	2009	316	18	35	2189	25
Intel Core 2 Duo P8700	2009	241	20	32	2254	25
Intel Core 2 Duo T9900	2009	530	15	25	2787	35
Intel Core 2 Duo SU9600	2009	289		60		10
Intel Core 2 Duo T9800	2009	530	15	28	2626	35
Intel Core 2 Duo SL9600	2009	316	21	43		17
Intel Core 2 Quad Q9000	2009	348	22	20	2863	45
Intel Core 2 Solo SU3500	2009	262	36	112	654	6
Intel Core i7-820QM	2009	546	14	19	3149	45
Intel Core i7-720QM	2009	364	16	17	3100	45
Intel Core i7-920XM	2009	1054	13	11	3797	55
Intel Core i7-740QM	2009	378	14	16	3318	45
Intel Celeron M U3400	2010	134	40	62	988	18
Intel Core i5-540M	2010	257	15	19	2826	35
Intel Core i5-520M	2010	225	16	20	2729	35
Intel Core i5-560M	2010	225	14	18	3116	35
Intel Core i5-560UM	2010	250	22	61	1976	18
Intel Core i5-520UM	2010	241	25	25	1492	18
Intel Core i5-580M	2010	266	13	16	3174	35
Intel Core i7-640UM	2010	305		26		18
Intel Core i7-620UM	2010	278	25			18
Intel Core i7-840QM	2010	568	13	16	3532	45
Intel Core i7-620M	2010	332	13	18	3044	35
Intel Core i7-940XM	2010	1096	12	13	4064	55
Intel Core i7-620LM	2010	300	16	27	2358	25
Intel Core i7-640LM	2010	332	15	21	2539	25
Intel Core i7-640M	2010	346	12	17	3307	35
Intel Celeron T3300	2010	86	29			35
Intel Celeron T3500	2010	80	31	39	1760	35
Intel Celeron B710	2011	70	25	88	868	35
Intel Celeron B810	2011	86	25	45	1633	35
Intel Celeron 847	2011	134	36	80	993	17
Intel Celeron B800	2011	80	26	47	1534	35
Intel Core i5-2540M	2011	266	12	17	3682	35
Intel Core i5-2520M	2011	225	12	18	3542	35
Intel Core i5-2537M	2011	250	17	25	2400	17
Intel Core i5-2557M	2011	250	14	24	2750	17
Intel Core i7-2637M	2011	289	14	22	2834	17
Intel Core i7-2760QM	2011	378	11	8	6001	45
Intel Core i7-2920XM	2011	1096	10	8	6131	55
Intel Core i7-2617M	2011	289	15	24	2762	17
Intel Core i7-2720QM	2011	378	11	10	5616	45
Intel Core i7-2620M	2011	346	11	16	3827	35
Intel Core i7-2960XM	2011	1096	10	7	6820	55
Intel Core i7-2640M	2011	346	11	15	3927	35
Intel Core i7-2657M	2011	317	14		2546	17
Intel Core i7-2860QM	2011	568	10	8	6323	45
Intel Core i7-2820QM	2011	568	11	10	5819	45
Intel Core i7-2677M	2011	317	13	20	2729	17
Intel Celeron B830	2012	86	23			35
Intel Celeron 887	2012	86	28	48	1414	17
Intel Celeron B815	2012	86	25	44	1645	35
Intel Core i5-3360M	2012	266	11	16	3995	35
Intel Core i5-3427U	2012	225	14	20	3180	17
Intel Core i7-3720QM	2012	378	10	8	6642	45
Intel Core i7-3920XM	2012	1096	9	6	6973	55
Intel Core i7-3667U	2012	346	11	18	3595	17
Intel Core i7-3820QM	2012	568	10	7	6849	45
Intel Core i7-3740QM	2012	378	10	7	6837	45
Intel Core i7-3520M	2012	346	10	15	4134	35
Intel Celeron 1017U	2013	86	24	46	1719	17
Intel Celeron 1000M	2013	86	21	42	1923	35
Intel Celeron 1037U	2013	86	22	41	1903	17
Intel Core i5-4300M	2013	225			4256	37
Intel Core i5-3437U	2013	225	13	20	3404	17
Intel Core i5-4300U	2013	287	13	19		15
Intel Core i7-3540M	2013	346	10	15	4320	35
Intel Core i7-4800MQ	2013	378	10	9	7232	47
Intel Core i7-4600U	2013	398	12	17	3216	15
Intel Core i7-4650U	2013	454	12	18	2974	15

Notes: N=170

Table A12: Summary for Online Retail Price Dataset

Model	Introduction	Intro Price	Exit Price	SuperPI1M	wPrime32	3DMark06	TDP(Watt)
AMD Athlon 64 3200+	2003	190	190	50	86		62
AMD Athlon 64 2800+	2004	173	120	52	95		35
AMD Athlon 64 3000+	2004	163	163	48	86		35
AMD Athlon 64 3400+	2004	191	207	45	85		62
AMD Athlon 64 3700+	2004	279	82	46	78		81
AMD Sempron 2600+	2004	63	49	60	109		25
AMD Sempron 2800+	2004	88	47	56	99		62
Intel Pentium 4 520	2004	169	176	60	120		91
AMD Athlon 64 4000+	2005	368	300	39	66		62
AMD Sempron 3000+	2005	98	28	47	93		35
AMD Sempron 3100+	2005	129	60	55	95		62
AMD Sempron 3300+	2005	134	88	52	84		62
Intel Pentium 4 560	2005	501	392	38	85		88
Intel Pentium 4 630	2005	240	159	44	103		95
Intel Pentium 4 660	2005	634	400	33	84		115
Intel Pentium D 820	2005	270	85	45	63	1412	95
AMD Sempron 3200+	2006	30	30		113		25
AMD Sempron 3400+	2006	135	39	41	95	664	62
AMD Sempron 3500+	2006	108	49	50	85	704	25
Intel Core 2 Duo E6700	2006	580	323	19	34	2354	65
Intel Core 2 Extreme X6800	2006	1100	982	17	31	2568	75
AMD Sempron 3600+	2007	121	41	49	86	748	25
AMD Sempron 3800+	2007	58	53	98			31
Intel Core 2 Duo E6600	2007	333	231	20	36	2052	65
Intel Core 2 Duo E6850	2007	279	160	17	29	2661	65
Intel Core 2 Extreme QX6700	2007	989	928	19		4022	130
Intel Core 2 Extreme QX6850	2007	1184	900	17		4450	130
Intel Core 2 Quad Q6600	2008	864	206	18	14	3547	105
Intel Core 2 Quad Q9550	2008	325	265	13	12	4230	95
Intel Core i7-920	2008	292	135	10	7	4728	130
Intel Core i7-940	2008	569	225	13	8	5054	130
Intel Core i5-750	2009	204	118	13	12	4320	95
Intel Core i7-720QM	2009	55	40	16	17	3100	45
Intel Core i7-740QM	2009	90	55	14	16	3318	45
Intel Core i7-820QM	2009	198	175	14	19	3149	45
Intel Core i7-920XM	2009	453	390	13	11	3797	55
Intel Core i7-950	2009	558	275	12	7	5143	130
Intel Core i7-960	2009	577	273	12	8	5360	130
Intel Core i7-975	2009	990	453	12	7	5837	130
Intel Core i3-330M	2010	103	22	21	24	2230	35
Intel Core i3-350M	2010	109	35	19	23	2366	35
Intel Core i5-430M	2010	154	30	18	21	2584	35
Intel Core i5-520M	2010	184	99	16	20	2729	35
Intel Core i5-520UM	2010	245	248	25	25	1492	18
Intel Core i5-540M	2010	184	99	15	19	2826	35
Intel Core i7-620LM	2010	304	307	16	27	2358	25
Intel Core i7-620UM	2010	282	286	25			18
Intel Core i7-640LM	2010	315	318	15	21	2539	25
Intel Core i7-640UM	2010	293	296		26		18
Intel Core i7-840QM	2010	297	297	13	16	3532	45
AMD A8-3850	2011	140	96	21	14	4027	100
Intel Core i5-2400	2011	188	150	10	12	5715	95
Intel Core i5-2500K	2011	225	180	9	9	5853	95
Intel Core i7-2600K	2011	330	259	9	6	6667	95
Intel Core i7-2617M	2011	397	397	15	24	2762	17
Intel Core i7-2620M	2011	188	165	11	16	3827	35
Intel Core i7-2630QM	2011	175	145	13	10	5039	45
Intel Core i7-2635QM	2011	448	448	14	15	4906	45
Intel Core i7-2640M	2011	238	188	11	15	3927	35
Intel Core i7-2657M	2011	397	397	14		2546	17
Intel Core i7-2720QM	2011	406	441	11	10	5616	45
Intel Core i7-2760QM	2011	401	392	11	8	6001	45
Intel Core i7-2820QM	2011	397	299	11	10	5819	45
Intel Core i7-2860QM	2011	590	590	10	8	6323	45
Intel Core i7-2920XM	2011	1003	553	10	8	6131	55
Intel Core i7-2960XM	2011	1372	1372	10	7	6820	55
AMD A10-5800K	2012	130	110	24	16	4464	100
AMD A4-5300	2012	60	41	28	36	2284	65
AMD A8-5600K	2012	110	92	25	17	4295	100
Intel Core i3-3110M	2012	109	109	16	22	2988	35
Intel Core i3-3120M	2012	123	123	15	22	3057	35
Intel Core i3-3220	2012	126	116	12	15	4019	55
Intel Core i5-3210M	2012	155	155	13	19	3553	35
Intel Core i5-3320M	2012	230	230	12	17	3767	35
Intel Core i5-3470	2012	190	172	10	12	6179	77
Intel Core i5-3550	2012	215	192	10	10	6405	77
Intel Core i5-3570K	2012	235	208	8	8	6561	77
Intel Core i7-2670QM	2012	299	150	12	11	5401	45
Intel Core i7-2700K	2012	377	308	8	6	6837	95
Intel Core i7-3610QM	2012	266	190	11	9	6078	45
Intel Core i7-3630QM	2012	234	234	11	8	6392	45
Intel Core i7-3720QM	2012	401	332	10	8	6642	45
Intel Core i7-3770K	2012	317	316	8	6	7229	77
Intel Core i7-3820QM	2012	590	575	10	7	6849	45
Intel Core i7-3960X	2012	1005	945	10	4	8330	130
Intel Core i5-3230M	2013	174	174	12	18	3760	35
Intel Core i7-4700MQ	2013	250	250	11	8	6883	47
Intel Core i7-4770K	2013	335	335	8	7	8030	84

Notes: N = 88.

How to Benchmark A Microprocessor

This section documents the rudimentary benchmarking experience by the author and includes some explanation for the benchmarks.

Benchmark Intel Core i5-3210M

I install SuperPI1M v 1.5, released on December 10^{th}, 2011. I benchmark the same microprocessor twice and the results are 13.796 seconds and 13.786 seconds. The benchmark results are consistent across different versions based on feedbacks from hobbyists.

Figure A1: Sample SuperPI1M Benchmarking Results

```
   1M Calculation Start.  19 iterations.
Real memory           =-417091584
Available real memory =1365073920
Allocated memory      =  8388648
  0h 00m 00.234s The initial value finished
  0h 00m 00.849s Loop 1 finished
  0h 00m 01.536s Loop 2 finished
  0h 00m 02.216s Loop 3 finished
  0h 00m 02.904s Loop 4 finished
  0h 00m 03.585s Loop 5 finished
  0h 00m 04.262s Loop 6 finished
  0h 00m 04.958s Loop 7 finished
  0h 00m 05.685s Loop 8 finished
  0h 00m 06.364s Loop 9 finished
  0h 00m 07.043s Loop 10 finished
  0h 00m 07.736s Loop 11 finished
  0h 00m 08.407s Loop 12 finished
  0h 00m 09.094s Loop 13 finished
  0h 00m 09.783s Loop 14 finished
  0h 00m 10.509s Loop 15 finished
  0h 00m 11.187s Loop 16 finished
  0h 00m 11.848s Loop 17 finished
  0h 00m 12.482s Loop 18 finished
  0h 00m 13.084s Loop 19 finished
  0h 00m 13.796s PI value output -> pi_data.txt
```

I install wPrime v 2.09, released on April 7^{th}, 2012. I benchmark the same microprocessor twice and the results are 19.446 seconds and 19.403 seconds. Since the microprocessor has 4 threads, the benchmark sets the number of threads to be 4 as shown in the screenshot. Hobbyists mention different versions of wPrime might affect the results. I assume the variations are small.

Super PI

SuperPI1M is a computer program that calculates π to one million digits after the decimal point. It uses Gauss-Legendre algorithm and is programmed firstly by Yasumasa Kanada in 1995 to compute π to 2^{32} digits. The method is based on the individual work of Carl Friedrich Gauss

Figure A2: Sample wPrime32 Benchmarking Results

```
Calculating sqrts of first 33554431 numbers.
Thread 1 calculating from 0 to 8388608 in THREAD#4F0.
Thread 2 calculating from 8388608 to 16777216 in THREAD#698.
Thread 3 calculating from 16777216 to 25165823 in THREAD#FD4.
Thread 4 calculating from 25165823 to 33554431 in THREAD#12E0.
Thread 1 is 25% complete @ 4.886 seconds
Thread 2 is 25% complete @ 5.046 seconds
Thread 3 is 25% complete @ 5.532 seconds
Thread 4 is 25% complete @ 5.623 seconds
Thread 2 is 50% complete @ 9.586 seconds
Thread 1 is 50% complete @ 9.682 seconds
Thread 3 is 50% complete @ 10.171 seconds
Thread 4 is 50% complete @ 10.252 seconds
Thread 2 is 75% complete @ 14.309 seconds
Thread 1 is 75% complete @ 14.331 seconds
Thread 3 is 75% complete @ 14.816 seconds
Thread 4 is 75% complete @ 14.855 seconds
Thread 1 complete with 0 errors!
Time taken: 18.966 seconds
Thread 2 complete with 0 errors!
Time taken: 18.993 seconds
Thread 3 complete with 0 errors!
Time taken: 19.36 seconds
Thread 4 complete with 0 errors!
Time taken: 19.39 seconds
Process completed in 19.403 seconds.
Test completed successfully.
```

(1777-1855) and Adrien-Marie Legendre (1752-1833) combined with modern algorithms for multiplication and square roots.

<u>wPrime</u>

wPrime32 is a multi-threading benchmark that calculates square roots with a recursive call of Newton's method for estimating functions. To find the square root of k, equivalently we solve for $x^2 = k$. The estimating function takes the form of $f(x) = x^2 - k$, where k is the number to be square rooted. Its first-order derivative is then $f'(x) = 2x$. The estimation starts with an initial guess $x_0 = k/2$ and calculate the sequence $\{x_i\}$ by Newton's method where $x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$ till $f(x) = 0$, i.e. $\frac{f(x_i)}{f'(x_i)} = 0$ and $x_{i+1} = x_i$. Then x_i is the accurate solution. wPrime repeats this method for all numbers from 1 to the 32 million integer.

<u>3DMark</u>

The description of 3DMark06 is based on the whitebook for 3DMark11.[30] 3DMark06 focuses on updating and rendering complex game worlds in real-time using DirectX 6. DirectX is a collection of application programming interfaces (APIs) for handling tasks related to multimedia, especially game programming and video, on Microsoft platforms. The benchmark workload consists of six tests. The final score is a weighted average of scores on the six tests. The raw score on each test

[30] Available at `http://www.3dmark.com/wp-content/uploads/2010/12/3DMark11_Whitepaper.pdf`.

is the number of frames per second.

Other similar graphics performance benchmarks are: 3DMark11 and Cinebench. Although the recent versions of the benchmarks can reflect differentiation in performance more accurately, they might also underrate older microprocessors. For example, since 3DMark11 uses DirectX 11, microprocessors designed in the context of DirectX 6 might underperform. This discrepancy also complicates interpolation for missing benchmark scores on old microprocessors.

Appendix 3: Relavent Information on Microprocessors

This appendix includes additional information on microprocessors in 2004-2013. I briefly introduce the main technological specifications I mentioned.

Number of Transistors

The transistor count of a device is the number of transistors in the device. Transistor count used to be the most common measure of microprocessor quality. According to Moore's Law, the transistor count of the integrated circuits doubles every two years. The first commercially available microprocessor, Intel 4004, had 2,300 transistors. Now there are commonly millions of transistors on a microprocessor. Transistor count is no longer a determinant microprocessor feature.

Lithography

Lithography means printing from a stone literarily. Since microprocessors, consisted of millions of transistors, are "printed" by ultraviolet lasers on silicon discs (called wafers), lithography became a term for the size of the transistors. This property of microprocessor is important because by halving the size of the transistors, the industry can easily abide by the Moore's law. Moore's law is the observation that the number of transistors on a micro processor doubles approximately every two years. The law is used in the semiconductor industry to guide long-term planning, such as Intel's 18-month "tick-tock" model.

Clockspeed

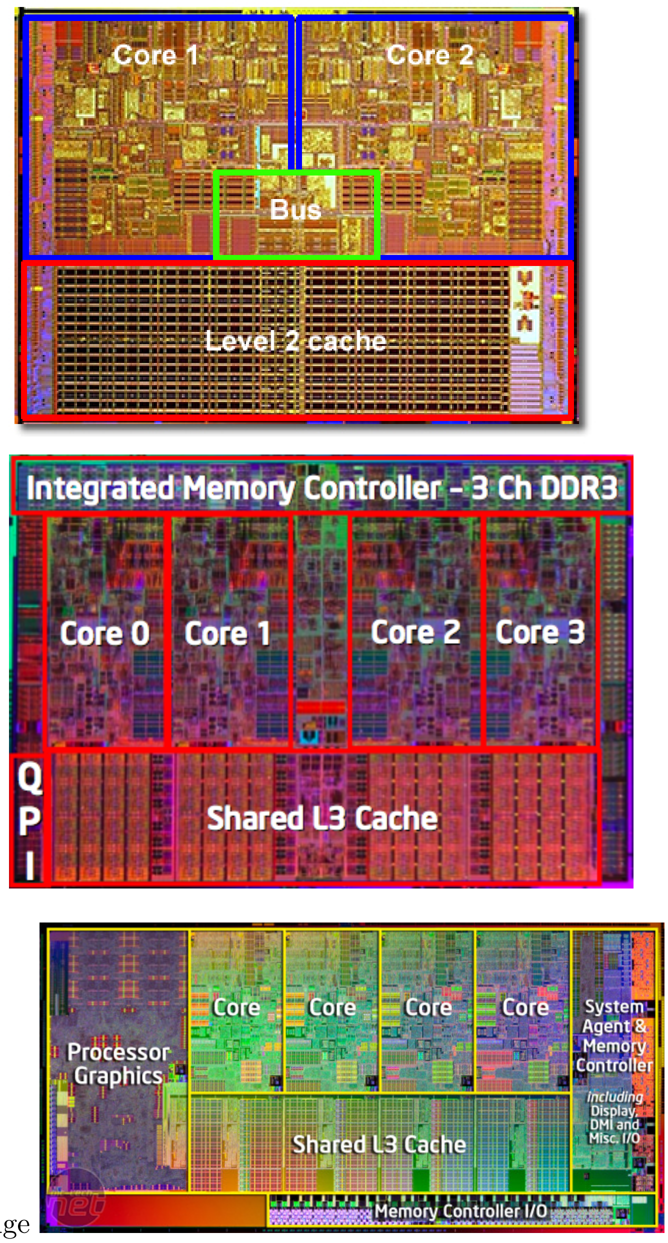
Clockspeed is the frequency at which a microprocessor is running, measured in Hertz. The way microprocessor processes an instruction is by completing a multi-stage "pipeline", in which each stage represents the completeness of the instruction. Each pipeline state takes on clock cycle to complete, so a smaller clock cycle (higher clockspeed) usually means more instructions processed per second.

Architecture

Architecture of a microprocessor is the layout design of components in the microprocessor.

An efficient design determines the microprocessor's speed and heat dissipation. The die maps best present different architectures as shown in Figure A3. These are among the Intel architectures mentioned in Table 6. AMD's architectures remain unexplored by the author.

Figure A3: Comparison of Architectures



Core

Nehalem

Sandy Bridge

Notes: Source: Intel Corporation.