6-2002

Further Examination of Moore's Law with Data Envelopment Analysis

Timothy R. Anderson
Portland State University, tim.anderson@pdx.edu

Rolf Färe
Portland State University

Shawna Grosskopf
Portland State University

Lane Inman
Portland State University

Xiaoyu Song
Portland State University

Further Examination of Moore's Law with Data Envelopment Analysis

TIMOTHY ANDERSON, ROLF FÄRE, SHAWNA GROSSKOPF, LANE INMAN and XIAOYU SONG

Address for Correspondence:
Timothy R. Anderson, Portland State University,
PO Box 751, Mail Code ETM, Portland, OR 97207-0751,
tima@etm.pdx.edu,
(503) 725-4668

Further Examination of Moore's Law with Data Envelopment Analysis

TIMOTHY ANDERSON, ROLF FÄRE, SHAWNA GROSSKOPF, LANE INMAN and XIAOYU SONG

ABSTRACT

Although it has stood the test of time for over 30 years, Moore's Law addresses but a single aspect of microprocessor design. As a proxy for technology, the number of transistors in an integrated circuit represents a limited perspective on the technology as a whole. Anderson *et al.* proposed a set of metrics by which to measure a technology, and a means to measure its progress over time utilizing data envelopment analysis. In this revised model, the assumption of state of the art (SOA) on product release is dropped, technical progress is measured iteratively over time, the effective time elapsed between the SOA and a no longer SOA has been refined to include a weighted average, and a means of utilizing proxy DMUs was implemented to maintain the dataset over time.

1. Introduction

In 1965, one of the founders of Intel, Gordon Moore, wrote on the future of integrated circuits, predicting the annual doubling of their components over the following decade [1]. Although initially discussed with respect to integrated circuits, this has become known as Moore's law and has been validated and expanded to include a wide array of computer systems and microprocessor performance. This rule of thumb has held with surprising accuracy over the last 36 years, with a gradual lengthening of the doubling period to 18-24 months.

As the components of integrated circuits improved over time, so too has the ability to assess their performance. Recent advances in performance measurement allow us to better capture the changing nature and sophistication of the circuit technology. For example, although the number of transistors in microprocessors is an obvious and simple measure of progress, it is not the only measure of technological advancement. Over time, microprocessors have been enhanced through an array of other features. In other words, the measurement of technology for microprocessors is not limited only to transistor count, but also includes other vital elements that provide useful and marketable solutions for computer users.

A large portion of traditional forecasting lends itself to single attribute extrapolation through time series or regression analysis to assess progress of a single attribute over time. To address the multi-factored nature of technology, one must present a quantitative combination of performance factors and physical characteristics to appropriately measure the state of the art, SOA [2, 3]. Practitioners have utilized technology indices, multiple linear regression, and even factor analysis to measure the advancement of technology to predict the rate of technological change and target reasonable futures based on a set of historical data [2, 4-10]. Indices typically utilize expert opinion to generate the SOA as a weighted function of relative performance or physical characteristics [11]. However, consideration should be given to the aggregation of attributes over keeping those attributes separated [5, 6, 12]. This methodology was further enhanced to include both functional and structural measures to further evaluate computer technology over time. Here structural measures may be considered physical characteristics of the technology, while functional measures pertain to that which the

technology delivers [13]. Additional considerations are those of major innovations or merely gradual improvements over time. These gradual improvements typically denote a continuous process of improvement of technologies whereas major innovations are represented as major disruptions. It should be noted, however, that often times major disruptions tend to smooth out over time as their impact is diminished [14].

In response to this limitation, Anderson *et al.* first introduced the utilization of data envelopment analysis (DEA) to measure the advancement of a multidimensional SOA surface over time [15]. This model was then extended to assess the overall advancement of relational database management system benchmarks [16]. The aim of this work is to combine these two earlier efforts to provide a robust methodology for multi-dimensional technological forecasting which takes into account the day-to-day tradeoffs that designers must face in a dynamic industry. Our approach allows us to take advantage of both observations on the SOA surface as well as those below it.

2. Data Envelopment Analysis (DEA) to Assess Technical Change

Since its initial introduction in 1978, DEA has been used for a wide variety of applications and cited in over 1500 articles [17]. Initially introduced and dubbed DEA by Charnes, Cooper, and Rhodes, this approach has been used extensively to assess performance in such diverse areas as education, health care, and banking [18]. Researchers have also used it to compare various products in a market, including computer printer rating, robotics, microcomputers, the financial performance of computer companies, and automobiles [19-23]. Each of these studies focused on product performance at a single point in time and not the change in performance over time. The contribution in Anderson *et al.* was to extend the use of DEA to assess the technological

rate of change over time. This differs from traditional dynamic DEA in that it does not use year-to-year windowing or productivity indexes, which require repeated observations over time, which are not available in the integrated circuit case.

The basic input-oriented DEA model with constant returns to scale is the basis of our approach and is represented by (1).

$$\begin{aligned} &\min \theta, \\ &s.t. \quad \sum_{j=1}^{n} x_{i,j}\lambda_j \le \theta x_{i,0}, \quad \forall i \in \{1,\ldots,m\} \\ &\qquad \sum_{j=1}^{n} y_{r,j}\lambda_j \ge y_{r,0}, \quad \forall r \in \{1,\ldots,s\} \\ &\qquad \lambda_j \ge 0, \forall j \end{aligned} \tag{1}$$

This is a linear programming problem, which must be solved for each observation being evaluated. The inputs and outputs of the observation under evaluation are denoted as $x_{i,0}$ and $y_{r,0.}$ Here, x and y correspond to the inputs and outputs of the environment. Specifically, $x_{i,j}$ refers to the *i'th* input of observation j and $y_{r,j}$ refers to the *r'th* output of observation j. These are the observations that constitute the sample and they serve in conjunction with the lambdas to construct the benchmark technology. The variable λ_j indicates the portion of system observation j used in setting a performance target for the current observation. Thus the solution value of θ for a given observation represents the production efficiency of the observation with respect to all the other observations, i.e., how close it is to the best practice of SOA.

The formulation given in (1) does not restrict the tradeoffs between inputs or between outputs. This formulation permits each decision making unit analyzed to use the tradeoffs, which maximize its performance relative to its peers. This flexibility is appropriate where products serve many different niche markets such as is the case with

microprocessors. In instances where restrictions in the acceptable tradeoffs are desired, there are a number of techniques that permit their implementation [24]. This model uses a constant returns to scale technology, which corresponds to a strong assumption of scalability where doubling a set of inputs results in a doubling of outputs. This assumption can be relaxed using a variety of DEA models but requires a larger data set.

3. Assessing Technical Rate of Change

Typically, DEA is used to assess performance during a fixed point in time. There are DEA models which assess the evolution of efficiency over time using moving time windows and Malmquist productivity indexes, but these methods require multiple observations of the same unit over a number of periods, and individual performance will change over time [25]. For example, annual data is collected on fifty hospitals over five years, 250 observations in all, to assess their relative efficiency and the change in efficiency over time. Unlike a hospital, a microprocessor does not fundamentally change performance over time. Barring an end-user's choice to overclock a microprocessor, a 166 Megahertz microprocessor in 1996 will always be a 166 Megahertz microprocessor.

In their original work, Anderson, *et al.* applied the model to all processors at once and evaluated the technical change over time based on the assumption that a microprocessor was deemed efficient on release with a constant rate of technical change [15]. Due to a variety of reasons including underperformance, delay to market, or model characteristics, the assumption of a product being SOA upon release may not hold and as such was relaxed in a later paper [16]. Processors that do not prove to be SOA upon release are projected to the SOA frontier using a combination of their reference observations and their relative weights, lambda's, in determining the processors' DEA

efficiency scores. As time progresses, additional SOA surfaces are generated and the advancement of this surface is determined through the assumption of constant technical progress as summarized in (2).

$$\theta_\tau = (1-\gamma)^\tau \theta_0 \qquad (2)$$

In this equation, θ_0 represents the time of release efficiency of the processor and θ_τ represents the efficiency at time τ. To calculate the technical progress over time, we solve for γ which results in θ_0 of 1. In an environment of progress, θ_τ will decrease over time as it is made obsolete by newer technologies. The coefficient of technological change, β, can be better represented by:

$$(1-\gamma) = \beta \qquad (3)$$

Substituting (3) into (2) yields (4).

$$\theta_\tau = \beta^\tau \theta_0 \qquad (4)$$

Since an individual observation's efficiency may be determined by multiple observations that occur at different times, the effective time passed since the setting of the relative efficiency frontier is determined through the use of those observations that are included in the evaluated observation's reference set. This effective rate of time may be denoted by:

$$\tau_{k,effective} = \frac{\sum_{j=1}^{n} (d_j - d_k)\lambda_{k,j}}{\sum_{j=1}^{n} \lambda_{k,j}} \qquad (5)$$

The effective time since release, $\tau_{k,effective}$, is calculated by taking the weighted average of the time passed since the present SOA was achieved. In this equation, d_j and

d_k represent the release dates of a reference SOA DMU j and the release date of the no longer efficient DMU k, d_k. The weight of the reference observation *j* on the efficiency score of observation k is denoted by $\lambda_{k,j}$. Calculating this rate of change on an iterative basis, one may also track the rate of change over time and see if it is indeed constant, or if it follows some other pattern.

Due to limitations in data collection, release dates are given as years rather than by month or specific date. This results in some measurement error in estimating a rate of change in that a processor released in January may be compared to the SOA a processor for December of the same year.

4. Proxy DMU Configuration

The earlier model of technical progress in Anderson, *et al.* called for the inefficient or non-state of the art observations to be dropped when they were deemed inefficient on release [16]. This works when there are numerous observations on the SOA surface but results in diminishing data sets in the event of significant numbers of non-SOA observations. To maintain the sample set, reference observations can be used to project inefficient observations to the SOA surface, using (6) for each inefficient observation at the time of release.

$$\begin{aligned} \hat{x}_{i,o} &= \sum_{j}^{n} \lambda_j x_j \forall i \in \{1..m\} \\ \hat{y}_{r,o} &= \sum_{j}^{n} \lambda_j y_j \forall r \in \{1..s\} \end{aligned} \tag{6}$$

This allows for the DMUs to be replaced by efficient representations of themselves determined by their SOA peers. This only occurs on the date of introduction, as we are interested in the frontier as it moves.

5. A Basic Example

To illustrate the model, a simple two-dimensional example will be presented using a small subset of the overall data, as seen in Table 1~~Table 1~~Table 1.

Obs	Processor	SPEC95 Int	SPEC95 Fp	Size (mm^2)	Power (W)	1/Tech (mm^{-1})	Log_{10} (Tran)	199X
A	Alpha 21064A	5.2	6.3	138	33	2	6.5441	4
B	Alpha 21164	8.5	12.7	209	50	2	6.9684	4
C	Intel P6 -200MHz	8.2	6.8	195	35	2.8571	6.7403	5
D	Sparc Ultra II	10.4	15	132	25	2.8571	6.4771	6
E	Power PC 603e -240MHz	6.3	4.6	197	5.5	2.8571	6.5563	6

Table 1 - Subset of Data

In order to illustrate with a figure, we focus on a simplified model: namely, integer performance to power ratio. The Standard Performance Evaluation Corporation, SPEC, generated the SPEC95 suite of benchmark tests to accurately measure the nature of processor performance. These tests evaluate the integer and floating-point operations using a given suite of benchmark tests in order to evaluate processor capability to run programs. The SPEC95 integer benchmark is plotted out versus power consumption in Figure 1~~Figure 1~~Figure 1. This illustrates the performance to power ratio over time and the shifts in the SOA surface.

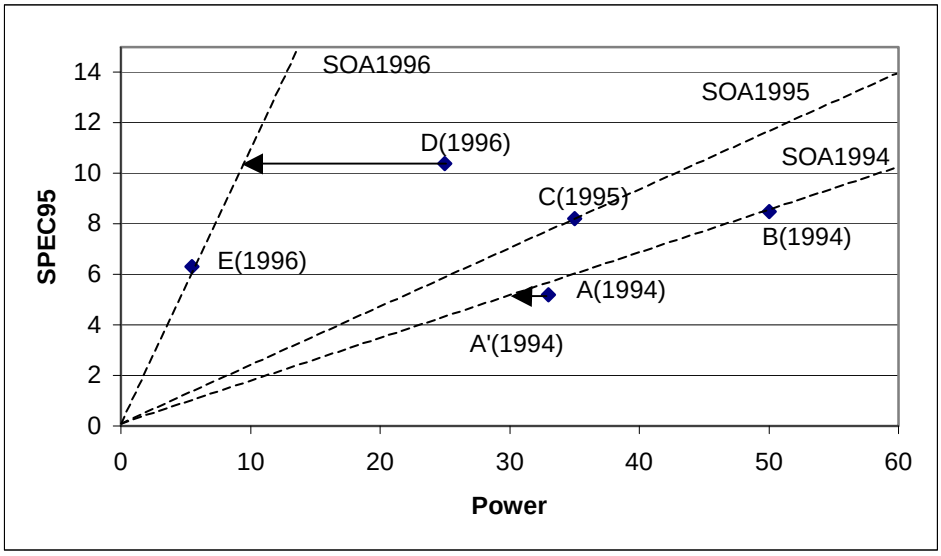

Figure 1 – Moving SOA Surface

Examination of the plot reveals that observation B defines the SOA surface in 1994, ahead of processor A. In this example, SOA is effectively set by the performance-to-power ratio. Simple calculation reveals that a performance to power ratio of 0.17 would require a power of 30.58 Watts to achieve the benchmark of 5.2. In terms of DEA efficiency this represents an efficiency score of 93%, i.e., the minimum expected inputs required for the given output are 93% of the actual inputs. Since it is not SOA, it is projected using the reference behaviors to an SOA proxy DMU of A' using (6) as presented in (7). The weight of reference behavior B, λ_B, used to determine the efficiency of A is 0.6116.

$$\begin{aligned} x &= 0.6116 \times 50 \\ y &= 0.6116 \times 8.5 \end{aligned} \qquad (7)$$

This reveals an A' with an input of 30.58 Watts which is used to calculate the coefficient of technical change over time. In 1995, processor C delivers a better performance to power ratio and thus the SOA is pushed forward, rendering observation B and A' no longer on the SOA frontier. Based on processor C's performance to power

ratio, observation B should produce the same level of performance using only 36.28 Watts or 72% of the 50 Watts used in 1994. Thus over the span of a year, the coefficient of technological change is 72%, corresponding to a rate of change of 28%. In 1996 the SOA advances significantly rendering observation C as no longer SOA. The efficiencies for A', B, and C at this point in time are calculated as 0.14, 0.14, and 0.20. Using (4) the coefficient of technological progress is calculated in (8).

$$0.14 = \beta^2(1.0) \qquad (8)$$

This results in a coefficient of technological change, β, of 0.38. Since processor C was SOA one year prior it has a coefficient of technological change of 0.20. In addition, D is not SOA and must be projected to determine a D' with a power input of 9.07 Watts as opposed to the 25 Watts that D currently uses.

6. Microprocessor Model

As previously mentioned, the measurement of microprocessor SOA should include more than the transistor count as in the original formulation of Moore's Law, or the power consumption used in the previous simplified example. Anderson *et al.* suggested the model presented in Figure 2~~Figure 2~~Figure 2.

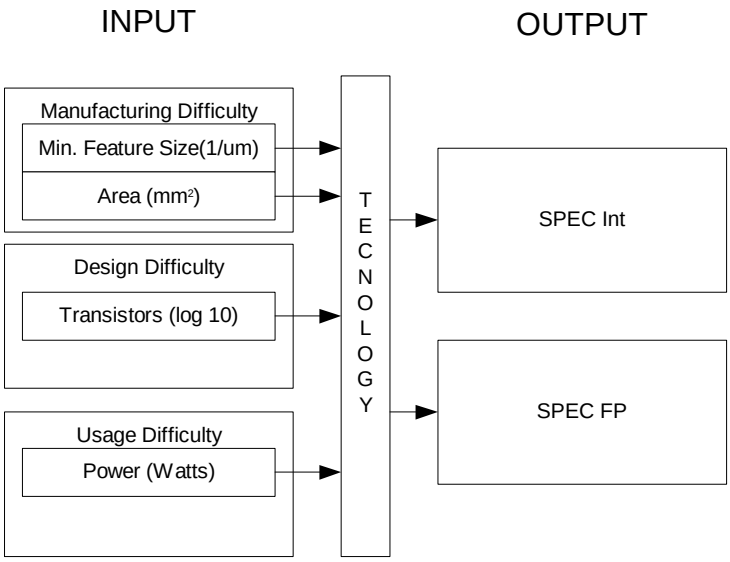

Figure 2 - Enhanced Microprocessor Technology Model

In the model, inputs reflect the difficulties associated with manufacturing, design and usage. Manufacturing difficulty is represented by minimum feature size and die area. This can be explained by the fact that smaller features and the larger die size of a microprocessor make high quantity production without yield problems more difficult. These factors are extremely important because the cost of silicon surface area has remained relatively constant over time, and thus the more that can be done with less space the better [26]. To address this, larger wafers have been adapted to maximize surface usage by the dies [27]. Combined with feature size reduction, wafer size allows for more transistors to be placed in the same area. Since 1971, manufacturing processes have become more difficult as the minimum feature sizes have been reduced from 10μm to 0.18μm due to the increased potential for defects caused by errant dust or other particles. Since the smaller feature sizes represent greater difficulty, the input is transformed through inversion to take into account that the closer to zero the feature size becomes, the more difficult the manufacturing process.

The number of transistors used represents design difficulty. Performance is often increased through increasing pipeline capacities, cache, or the number of registers, which are done by adding transistors. This increased pipeline capacity complicates testing requirements and all aspects of design. Because the number of transistors has increased at an exponential rate over time, this has been $\log_{10}$ transformed. Although there are other elements, which affect design difficulty, many of these issues, including the affects of multiple layers and wire resistance, are both directly and indirectly affected by the number of transistors.

Usability is reflected by the power consumption in Watts. Although often overlooked in desktop applications, its importance is significant. The increase in power consumption and clock speed increases heat generation, which has adverse effects on performance through increased wire resistance and other aspects. Additionally, as smaller and smaller devices become available, battery consumption and conservation are becoming increasingly important.

The model outputs are directly tied to the speed and ability of a processor to run programs. The faster a program can run, the larger demand that can be answered by computer software. Finding an appropriate means to measure this performance is difficult [7]. The earliest microprocessor performance measurement was millions of operations per second (MIPS) which was flawed as a measurement due to the ability to perform a million "no-operation" commands, which took clock cycles but not processing power. In 1988 the Standard Performance Evaluation Corporation (SPEC) was introduced for computer workstation performance evaluation. The SPEC CPU subcommittee benchmarks processor performance through two program suites designed

to evaluate floating point and integer arithmetic. The SPEC95 benchmark score is based on the geometric mean of the suite and then normalized against a SPARCStation 10/40. The previous version SPEC92 was similarly based but had a more limited test suite. In 1992, SPEC released SPEC92, which was followed by SPEC95 and has since been again enhanced for SPEC2000. The data for our study pertains to SPEC92 and SPEC95 over the 1990-1999 time period.

7. Results

Analysis of SPEC95 data reveal a slower than expected rate of technological progress than would be expected from Moore's Law. However, it also takes a broader perspective than Moore's Law by including more complex outputs and accounting for `inputs'. Results of SPEC95 data for the years 1992-1999 are illustrated in Table 2~~Table 2~~Table 2. It is interesting to note that as time passes, the 95% confidence interval decreases over time, which may be interpreted as better compliance with the expected rate of change.

Mean β:	0.842304
Standard Deviation:	0.076756
95% Conf:	+/- 0.022679

Table 2 - Rate of SOA Change

These results imply the doubling of the technology every 42-53 months rather than the 18-24 months proposed by Moore. Moore's rate would correspond to a β of 0.630 – 0.707, but only represents a single aspect of microprocessor technology. By combining the number of transistors with other factors, a more complex and complete picture of microprocessor technology and its advancement is provided. Part of the explanation of the slower rates of change can be illustrated through the minimum feature size. For example, SOA for feature size represented by the Pentium III Coppermine chip

was 0.15 micrometers in 1999 compared to 0.6 micrometers for the Sun SuperSPARC in 1992. Over seven years this factor decreased by 75%, indicating a doubling of difficulty every three and a half years, which corresponds to a much slower rate than that of transistor count. In addition the die size shrank from 315 to 106, which corresponds to a technological doubling every four and a half years. Individually these items help explain the reduced rates of technological change over time and provide a more detailed picture of this evolution. Figure 3~~Figure 3~~~~Figure 3~~ illustrates the movement of the technological rate of change and its 95% confidence interval over time. In 1994 there were too few data points to provide a 95% confidence interval, and once it is established the interval narrows.

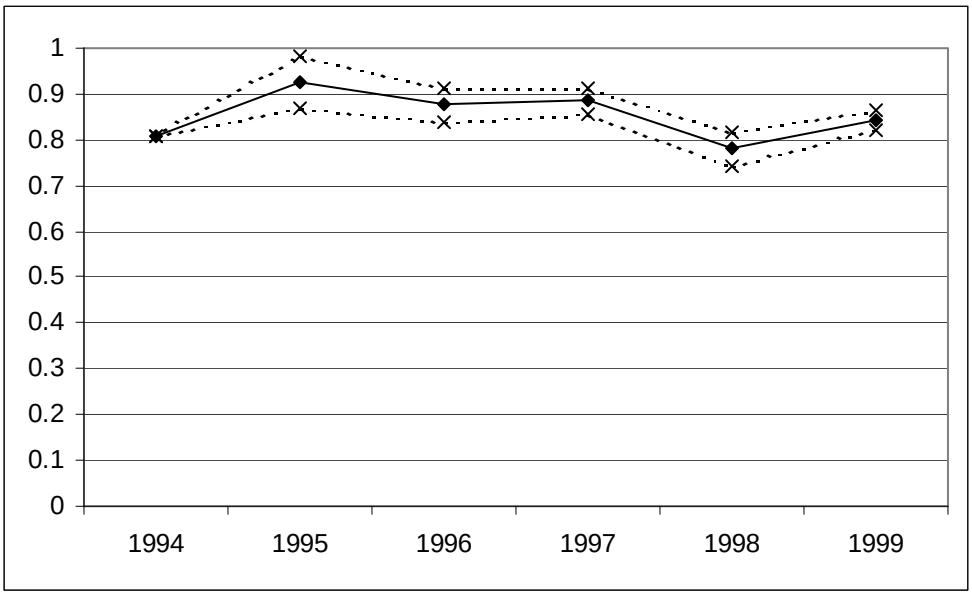

Figure 3 - SPEC95 Technological Rate of Change by Year

As Figure 3~~Figure 3~~~~Figure 3~~ illustrates, the movement of the SOA surface appears fairly consistent over the years with a slight increase in 1988. The large drop in efficiency in 1995 may be associated with a stabilizing of the system rather than an overall change. Some of the limitations of annualized data may also be evident, as they

do not give enough points in time to better establish a reference behavior.

Some insight into reference behavior may be obtained through the exploration of the SPEC92 benchmark data that served as the predecessor to SPEC95. Although significantly different, SPEC92 served as the model for SPEC95 and does share some commonality. Table 3~~Table 3~~Table 3 presents the overall results for SPEC92 for the years 1992-1996. The year 1996 represents the final year of the SPEC92, and most notably it lacks benchmark scores for one of the SPEC95 SOA processors, the Power PC 603e-240 MHz, which may explain the reduced rate of technical change in the final year of the benchmark.

Mean β:	0.902581
Standard Deviation:	0.080984
95% Conf:	+/- 0.028059

Table 3 - SPEC92 Technological Rate Of Change

The SPEC92 results for the years 1990-1996 are displayed in Figure 4~~Figure~~ 4~~Figure 4~~. Here, however, there is a much more notable decrease in the 95% confidence interval over time as the range further narrows to a point approaching that of the SPEC95 data in later years. This may indicate that the benchmark is maturing over time, and its acceptance is growing, allowing for larger samplings. These rates are similar to SPEC95, although a different benchmark may be considered at least partially when analyzing data for SPEC95, as this was a derivative of SPEC92.

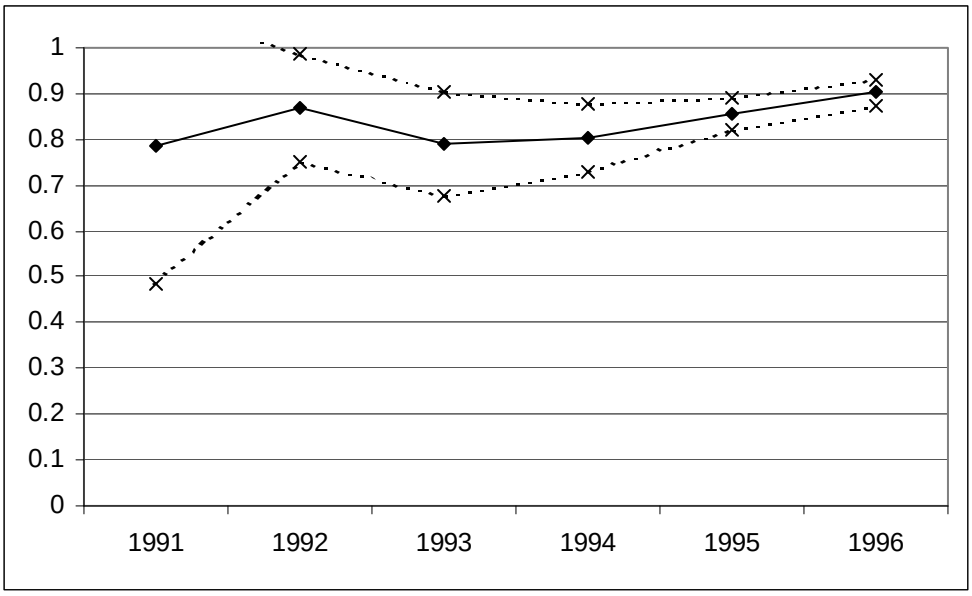

Figure 4 - SPEC92 Technological Rate Of Change

8. Discussion

Our modified DEA methodology provides a complement to traditional forecasting methods. It allows for dynamic trade-offs and permits the identification of key inflection points, which may be bettered, when targeted by development and engineering. The aim of this methodology is to provide more insight and detail into the forecasting of microprocessors over and above that which Gordon Moore predicted over 30 years ago. It is not meant to replace or contest this long established adage, merely to provide additional insight into the advancement of the technology. By taking older benchmarks and examining the rate of change, it may also be possible to extrapolate data to future derivatives of those benchmarks. For SPEC 2000 benchmarks, for example, one may be able to extrapolate that similar behavior will follow based on the fact that similar patterns occurred in the past. One could compare results with those of SPEC2000 over time and assess if the rate of change appears to be fairly consistent or whether it changes over time.

This approach could be expanded to include the forecasting and monitoring of other micro electro mechanical systems (MEMS) and their performance and attributes over time as well as many other fast-changing technologies. Of course, the application would be dependent on identification of key inputs and outputs. Additional insight could also be attained through further granularity of the data set beyond the current annualized dataset.

TIMOTHY R. ANDERSON is an Associate Professor in the Engineering and Technology Management Department at Portland State University. He has served in various positions including Program Chair, Program Co-Chair, and Director of Technical Activities for the Portland International Conference on the Management of Engineering and Technology in 1997, 1999, and 2001.

ROLF FÄRE and SHAWNA GROSSKOPF are professors of Economics at Oregon State University. Both have them authored or co-authored over 100 papers and a half dozen books in the fields of productivity analysis.

LANE INMAN is a Ph.D. student in System Science at Portland State University. He currently works at VERITAS Software as a Principal Consultant for their Enterprise Consulting Services where he specializes in Enterprise Storage Infrastructure.

XIAOYU SONG is an Associate Professor of Electrical and Computer Engineering at Portland State University specializing in circuit design and VHDL.

9. References

[1] G. E. Moore, "Cramming more components onto integrated circuits," *Electronics*, pp. 114-117, 1965.

[2] E. N. Dodson, "A General Approach to the Measurement of State of the Art and Technical Advance," *Technology Forecasting*, vol. 1, pp. 391-408, 1970.

[3] A. J. Alexander and J. R. Nelson, "Measuring Technological Change: Aircraft Turbine Engines," *Technological Forecasting and Social Change*, vol. 5, pp. 189-203, 1973.

[4] D. Sahal, "Foundations of technometrics," *Technological Forecasting and Social Change*, vol. 27, pp. 1-37, 1985.

[5] H. R. Rao and B. D. Lynch, "Hedonic Price Analysis of Workstation Attributes," *Communications of the ACM*, vol. 36, pp. 95-102, 1993.

[6] D. Sahal, "On the Conception and Measurement of Trade-Off in Engineering Systems: A Case Study of the Aircraft Design Process," *Technological Forecasting and Social Change*, vol. 8, pp. 371-384, 1976.

[7] D. V. Hall, *Microprocessors and Interfacing: Programming and Hardware*, 2nd ed. New York: McGraw-Hill, 1990.

[8] D. K. Peterson, P. E. Miller, W. A. Fischer, and R. W. Zmud, "Technology Measurement and the Appraisal of Information Technology," *Technological Forecasting and Social Change*, vol. 42, pp. 251-259, 1992.

[9] E. Esposito, "Technology Measurement: A Composite Approach," *Technological Forecasting and Social Change*, vol. 43, pp. 1-17, 1993.

[10] A. L. Porter, A. T. Roper, T. W. Mason, F. A. Rossini, and J. Banks, *Forecasting and Management of Technology*. New York: John Wiley & Sons, 1991.

[11] T. J. Gordon and T. R. Munson, "A Proposed Convention for Measuring the State of the Art of Products or Processes," *Technological Forecasting and Social Change*, vol. 20, pp. 1-26, 1981.

[12] D. Sahal, "The Generalized Distance Measures of Technology," *Technological Forecasting and Social Change*, vol. 9, pp. 289-300, 1976.

[13] K. E. Knight, "A Functional and Structural Measurement of Technology," *Technological Forecasting and Social Change*, vol. 27, pp. 107-127, 1985.

[14] J. P. Martino, *Technological Forecasting for Decision Making*, 3 ed: McGraw-Hill, 1992.

[15] T. R. Anderson, S. Grosskopf, R. Fare, and X. Song, "Examining Moore's Law Using Data Envelopment Analysis," *IEEE Journal on Engineering Management*, Under Revision, 2001.

[16] T. R. Anderson, K. Hollingsworth, and L. Inman, "Assessing the rate of change in the enterprise database system market over time using DEA," in *Technology Management in the Knowledge Era*, D. F. Kocaoglu and T. R. Anderson, Eds. Portland: PICMET, 2001, pp. 384-390.

[17] W. W. Cooper, L. M. Seiford, and K. Tone, *Data Envelopment Analysis : A Comprehensive Text with Models, Applications, References and DEA-Solver Software*: Kluwer Academic Publishers, 1999.

[18] A. Charnes, W. W. Cooper, and E. Rhodes, "Measuring the efficiency of decision making units," *European Journal of Operational Research*, vol. 2, pp. 429-44, 1978.

[19] J. Doyle and R. Green, "Strategic choice and data envelopment analysis: comparing computers across many attributes," *Journal of Information Technology*, vol. 9, pp. 61-69, 1994.

[20] S. Thore, F. Phillips, T. W. Ruefli, and P. Yue, "DEA and the management of the product cycle: The U.S. computer industry," *Computers & Operations Research*, vol. 23, pp. 341-356, 1996.

[21] J. R. Doyle and R. H. Green, "Comparing products using data envelopment analysis," *Omega*, vol. 19, pp. 631-8, 1991.

[22] M. Khouja, "The Use of Data Envelopment Analysis for Technology Selection,"

Computers and Industrial Engineering, vol. 28, pp. 123-132, 1995.

[23] C. L. Storto, "PICMET '97: Innovation in Technology Management: The Key to Global Leadership," presented at Portland International Conference on the Management of Engineering and Technology, Portland, OR, 1997.

[24] R. Allen, A. Athanassopoulos, R. G. Dyson, and E. Thanassoulis, "Weights restrictions and value judgements in data envelopment analysis: Evolution, development and future directions," *Annals of Operations Research*, vol. 73, pp. 13-34, 1997.

[25] R. Fare and S. Grosskopf, *Intertemporal Production Frontiers: With Dynamic DEA*: Kluwer Academic Publishers, 1996.

[26] G. E. Moore, "Intel-memories and the microprocessor," *Daedalus*, vol. 125, pp. 55-80, 1996.

[27] S.-M. Kang and Y. Leblebici, *CMOS Digital Integrated Circuits: Analysis and Design*, 2nd ed. New York: McGraw-Hill, 1999.