3D Performance Surfaces- A Tool for Analysing and Estimation of Production System Performances

Jan-Arne Pettersen¹, Anders Segerstedt², Børre Bang¹

¹ Narvik University College, Postboks 385,
8505 Narvik, Norway

janarpe@hin.no / bb@hin.no

² Luleå University of Technology, Industrial logistics,
S-97187 Luleå, Sweden

anders.segerstedt@ltu.se

Abstract

This paper presents a new method to describe, analyse and estimate production system performances. According to current literature, the basic performance measures are *WIP* (Work-in-Process), *lead time* and *throughput*. These are also in this article used as performance measures. Continual trade-offs between different performances are necessary to stay competitive in today's market. Different performances are archived by adjusting system parameters. Knowledge about relations between system parameters and system performances in existing systems, and about system performances of not yet implemented system alternatives, is essential for business. Queuing theory and simulation can estimate system performances of not yet implemented systems and help the decision makers, but when the complexity increases queuing theory becomes very difficult and heavy to use. A single simulation presents limited information. Multiple simulations are necessary to ensure that the best alternative is chosen. A high number of simulations demand a lot of computer time and resources. Reduction of runs is desirable even with cheaper computer equipment. Currently, traditional two-dimensional charts are the only tools to present and analyse system performances. This article presents a new surrogate model for easier estimation and presentation of system performances, their internal relations, and relations to the system parameters. With the new surrogate model, system performances based on simulations are presented as positions in a three dimensional environment. Parametric curves and surfaces of Bezier type are generated and adapted to these positions. System performances of other system alternatives are then estimated. The number of simulation calculations can thereby be moderated. The method is illustrated with a small production line system.

Keywords: Production Performance, Kanban, Simulation, Surrogate model

Introduction

In this article, a new spline (?) based surrogate model of a stochastic Kanban controlled production line is presented. The model is based on, and combined with, a simulation model. The main purpose of the surrogate model is to show the relation between multiple inputs and three responses, with easier estimates of system behaviour, based on interpolation of a few simulations conducted in a larger solution space. Barton (1998) presents a review of surrogate models or metamodels (different names, same topic).

If a production system is complex, or if there are several alternatives, the demand of simulation time and simulation resources becomes high. To reduce the simulation effort and to perform optimization, the literature describes the use of surrogate or metamodels. Yu and Popplewell (1994) and Barton (1998) describe and define metamodels for simulation. A surrogate, or metamodel, is a deterministic simplified model of the simulation model that describes the relation between input variables and responses. Kleijnen and Sargent (2000) determine the goals of metamodels to be: problem entity understanding, prediction, and optimization. In Barton (1998), multiple metamodel methods are described. These are response surfaces, splines, radial basis functions, neural networks, spatial correlation models, and frequency-domain approximations. In this article, only the use of splines is described, and then a new alternative spline method is introduced.

Modern production systems often include multiple processes in tandem. Stochastic process times lead to blockages and starvations, which reduce the process utilization, increase system lead time and decrease throughput. To stay competitive, companies need to perform trade-offs between performances by adjusting system parameters, like the size and location of buffers. To do this, companies need knowledge about the relations between system parameters and system performances in the existing system, and for not yet implemented system alternatives. From now on, in this article, the term *input* is used for the system parameters and *response* for system performances. In some cases the relations between inputs and responses can be shown and described analytically, but if the system is complex, as it often is due to multiple tandem stochastic processes, simulation is preferred (Köchel (2002)). To obtain a credible result, multiple random streams have to be tested. This means that each system configuration has to be analysed by multiple runs. Runs of one configuration give only information concerning that special configuration. To find the best configuration, all possible configurations have to be simulated and compared or a search routine has to be used to find the best or the near best configuration. Fu (2002) describes and analyses different search routines and optimization methods for simulation, with special focus on how the research in this domain and the development of commercial software are evolved in different directions. Our method is not an optimization or a search routine in itself, but gives a better picture of how the system would react when adjusting multiple parameters. The method is, as Fu (2002) addresses, a general method even if it is only exemplified with one example in this article.

The literature describes multiple production system performances (cf. Hopp and Spearman (2000), Silver and Pyke (1998), Rolstadås and Andersen (2000), Browne et al (1996), Lee and Wang (2007)).). The basis of most of them is *WIP* (Work-in-Process), *lead time [LT]* and *throughput [TH]*. The method described in this article focuses on these three basic production performances; knowledge about these is necessary before other performances can be analysed. Therefore, the responses *WIP* [items], *LT* [time units] and *TH* [items/time units] are used. The relation between them is described by Little's formula $WIP = LT \cdot TH$ (proved by Little (1961)). *WIP* is measured as the average number of items in the system during a time

period T , LT is measured as the average number of time units the items use through the system and, TH is measured as the average number of items per time unit delivered out of the system.

Existing tools for presentation and analysis of the three basic system performances are traditionally two-dimensional charts, see Muckstadt and Tayur (1995b), Gstettner and Kuhn (1996), Savsar and Al-Jawini (1995), Herer and Shalom (2000) etc. for examples. In these charts, performances are analysed two at a time. It is difficult to see the relation between them and the system parameters. Even in a system as shown in figure 4, where all items follow the same path with only five processes, it is difficult to see how system performances relate to different system alternatives.

Existing surrogate models are often represented as $y = g(x)$, where x is the input, y is the output or response and g is the function that represents a simplified relation between the input and the response (see figure 1) (Barton (1998)).

Figure 1: Existing surrogate models

A function like this can be expanded to $y = g(x_1, x_2)$, where x_1 and x_2 are two different inputs. For example, Kirkavak et al (1999) use this method. Multiple inputs to one response are included in the response surface method. However, to analyse more than one response, multiple functions are needed, for instance as $y_1 = g_1(x_1, x_2)$ and $y_2 = g_2(x_1, x_2)$ (Barton (2006)). With separate explicit functions, visualization, evaluation, and interpolation of combined responses are difficult. Our model overcomes some of these difficulties by operating in the parametric domain of the response functions.

For those not familiar with parameterisation and change of variables, a quick review is given as follows. The function $y = g(x)$ can be reformulated to a parametric representation by simple change of variables. Introduce u , and write

$$f(u) = (u, g(x))$$

where $x = u$ and $y = g(u)$. $x = u$ can further be replaced by $x = g_1(u)$ and together with $y = g_2(u)$, $f(u)$ becomes:

$$f(u) = (g_1(u), g_2(u))$$

$x = g_1(u)$ is still the input, $y = g_2(u)$ is the response, and u is the parametric value that some textbooks describe as the elapsed time of our journey along the curve. Note: x is now not necessary equal to u .

By further development, u is now the parametric value and set to be the input variable to the simplified system. Further, $x = g_1(u)$ is redefined from being the input variable to be response #1 and renamed to y_1 . y is renamed to y_2 . So, the two responses become response #1 = $g_1(u) = y_1$ and response #2 = $g_2(u) = y_2$ and can be expressed through the parametric function $f(u) = (g_1(u), g_2(u))$ with u as the input variable. By using this change of variables, one input can be related to two responses. This is graphically illustrated in figure 2. In figure 2, the input value u_5 , that is also the parameter value, gives $f(u_5) = (g_1(u_5), g_2(u_5)) = (y_{1a}, y_{2a})$. (y_{1a}, y_{2a}) is the coordinates of the solution point (position) connected to the input value u_5 and the values of response #1 and response #2.

Figure 2: New surrogate model with one input and two responses

This model can, in the same way, be expanded further with one more response and more inputs.

$$f(u, v) = (g_1(u, v), g_2(u, v), g_3(u, v))$$

Then the model handles two inputs and three responses. The relation between all elements can be illustrated in a 3-dimensional environment, as seen in figure 3.

Figure 3: New surrogate model with two inputs and three responses

The model can be expanded further with one more input w , but then it is difficult to illustrate visually on paper.

$$f(u, v, w) = (g_1(u, v, w), g_2(u, v, w), g_3(u, v, w))$$

This article presents decision alternatives in a stochastic Kanban controlled production line, a method to present the three basic production system performances, a surrogate model with multiple inputs and three responses, and a method to estimate and predict performances of not simulated system alternatives. There is literature that describes relations between system performances and system parameters in Kanban controlled production systems (cf. Tayur (1993), Muckstadt and Tayur (1995a) (1995b), Gstettner and Kuhn (1996), Pettersen and Segerstedt (2008) etc.); but mostly it can only work as guidelines. Real performances of a specific production system cannot be told from general rules. Each system has to be analysed specifically. Our method simplifies this.

Decision alternatives in a single-line-single-product-multi-process system

Figure 4 illustrates two control alternatives for a single line production system for a single product. In the system, there are buffers to reduce the effects of stochastic process times. In a system like this, there are two main decisions to be made: first, the control policy and second, the related system parameters. There are multiple control policies available. This article uses a Kanban policy to exemplify the method.

Figure 4. Two Kanban Inventory Control Methods

In a Kanban controlled system, cards or other signs are used to control the level and the position of system inventories. Each card is related to at least one buffer and at least one process. Each card is related to only one item at a time and an item is always connected to a card as long as it is in the system. The number of cards gives the maximum number of items inside a given region. An item can leave a current region only if there is a free card available in the next region. When an item leaves the region, the free card returns to the beginning of the region and a new item is allowed to enter. A card stays connected to the same item the whole time that item is in the given region.

The system parameters of a Kanban system are the number of cards in each region. It is important to be aware of two things: 1. The same system maximum inventory can be achieved with multiple card allocations among the regions. In this article, a specific card allocation is referred to as a system configuration. 2. The system parameter is the maximum inventory, in the overall system or in part of it, in contrast to the performance measure average WIP (cf. Pettersen and Segerstedt (2008)). A Kanban controlled system can be organized in many ways (cf. Muckstadt and Tayur (1995a, b), Hopp and Spearman (2000), Silver et al (1998)). This article uses the same Kanban Cell Control structure as in Tayur (1993) and in Muckstadt and Tayur (1995a,b). It is illustrated in figure 4. This is the same as the Immediate Material Transfer (IMT) Kanban system described in Gstettner and Kuhn (1996). (E.g. Pettersen and Segerstedt (2008) cover the topic Kanban Control where every buffer before the process is restricted.) The advantage of Kanban Cell Control is that a card is connected to the part in front of, inside, and after the process. In this way, the card, which symbolises a buffer space, is moved dynamically and automatically to the place where it is needed, in front or after the process. If a process stops or obtains a long process time, its own cards and the upstream process's cards, fill the buffers in front of the process. In figure 4 there are two buffers between two processes. These could be one physical buffer, where items connected to both cards, from the upstream process and the downstream process, could be stored together. Gstettner and Kuhn (1996) shows that IMT has better performances, throughput versus inventory, than a traditional Kanban system. We assume infinite delivery of raw material and infinite demand. In the same way as demonstrated in Tayur (1993), Muckstadt and Tayur (1995a), there is no need for raw material storage and there will never be no need for a buffer after the last process. In other words, the first and the last Kanban Cell (A and E in figure 4) should always have exactly 1 card. To exemplify our method, system performances of

different card allocations among cell *B*, cell *C* and cell *D* will, in this article, be simulated, analysed and estimated.

The 3D Performance Surface Method

System performances are generated mainly from production histories of existing systems or from artificial production histories generated by simulation. In this article, production histories generated with simulation are used, but the method would work in the same way with real production histories. In traditional analyses of system configuration, only two system performances are presented as points in a two-dimensional diagram. For example, inventory versus throughput, lead time versus inventory, or throughput versus lead time. One point in the diagram represents a specific system situation, specific control method and system configuration. Instead of using only two dimensions at a time, this method presents a Performance Point of a given configuration, as a position in the three dimensional space. The responses are placed in the space as followed: Average throughput (output rate) on the *x*-axis, average lead time on the *y*-axis, and average *WIP* on the *z*-axis (figure 6).

Figure 6. Performance Point with Standard Deviation Spheres presented in a 3D space

A Performance Point will always be near the surface, $WIP = LT \cdot TH$, generated by Little's formula (Little (1961)), which in this article is referred to as the Little's Surface. Little's Surface is gridded on figure 6. To indicate and visualise the variance in the three responses, a sphere is located at each Performance Point. The sphere is divided into three equal parts, one part for each response, with separate grey scaling, as indicated in figure 6. White indicates the highest value, black the smallest value. The variance measure can be the absolute value of the

standard deviation, a relative measure relating the standard deviation to the mean value of the corresponding response at that position, or a scaled measure relating to a scale given by the user, or a scale relating to the whole data set. So the use of the variance sphere is dependent on the analyser's needs. The range of the greyscale should be chosen with caution, and in a way that highlights the area of interest.

In this way, it is easier to evaluate and do trade-offs between all the system performances, including average and variance measures. No other presentation or analysis tool, as far as we are aware, can present six performance measures in the same picture. On the computer screen, it is possible to rotate and zoom the 3D model to do visual investigation of the performances. The benefit of this is difficult to show on paper (figure 6), so we can only appeal to the reader's imagination.

But what can be shown, is how it is possible to “navigate” around on the Little's Surface by adjusting the system variables (inputs) to archive different system performances and even estimate the performance without new simulation runs. These topics are addressed in the rest of this article.

Estimation of system performances

Until now, in this article, all the presented performance points are generated by simulation. Each configuration is simulated multiple times to generate credible performance data. Depending on system complexity, process performance distributions and demanded accuracy of results, each simulation demands computer time and resources. One way to find the best configuration among a set is to simulate all of them, but the number of configurations gets large, even for small and simple systems. In the following, a case of five Kanban Cells, illustrated in figure 4, is used to show the need for, and benefit of, a surrogate model to reduce the simulation time and resources.

In the five Kanban Cells case the goal is to obtain knowledge about how the system performs changes when the number of cards in Cell B is varied between 1 and 9, in Cell C between 1 and 9, and between 1 and 12 in Cell D. There is only one card in Cell A and one in Cell D. In this case, 972 configurations have to be evaluated and simulated. In a production system like this, with stochastic process data, each configuration has to be simulated over a given period of time, referred to as the run length, and a number of replications have to be performed. This is to gather sufficient artificial historical material to generate an adequate statistical measurement. The number of replications and the run length depends on the system. There are multiple studies in the literature regarding this issue (Banks, 2005). This study does not address this issue, but merely concludes that a simulation of a given configuration has to be ran for a given number of time units, and a number of replications have to be performed, to reach an acceptable confidence interval. The surrogate model, described in this article, reduces the number of simulated configurations. The benefits of this depend of the run length. For systems with long run length, the benefits are larger due to larger reduction of the total calculations.

For illustration purposes, 5 replications of each simulation are used. To obtain knowledge about all the configurations in the five Kanban Cells case, a total number of 4860 runs then has to be performed. In the following, a method to reduce this number to 720 is presented. It is the writers' belief, though this has not yet been tested, that the number could be reduced to 320.

From now on the terms *simulated* performance point and *estimated* performance point will be used on, respectively, performance points based on information from simulation, and performance points based on information from the surrogate model. Present literature

describes a number of interpolation methods. Geometrical Bezier interpolation is chosen as the surrogate model because it is quite easy to implement (Mortenson, 1997). Other interpolation techniques may be better, so the plan is to address this in future research. For a detailed description on the interpolation methodology see Mortenson (1997) and Farin (1993).

A Bezier surface patch is in the form $P(u, v) = \sum_{i=0}^n \sum_{j=0}^n P_{ij} B_{i,n}(u) B_{j,n}(v)$, where P_{ij} is the number of control points for $i=0, \dots, n$ and $B_{i,n}(u)$ and $B_{j,n}(v)$ is the so called Bernstein polynomials (here just shown for u):

$$B_{i,n}(u) = \binom{n}{i} u^i (1-u)^{n-i} \text{ with } \binom{n}{i} = \begin{cases} \frac{n!}{i!(n-i)!} & \text{if } 0 \leq i \leq n \\ 0 & \text{else} \end{cases}$$

It is important to notice that $u \in [0,1]$ and $v \in [0,1]$. In using Bezier interpolation and the parameter variable as input, as described in the introduction, a second degree curve representing the change in one input can be generated from only four points. Linear, or first degree, would not give sufficient accuracy, and third degree has the possibility of fluctuating (Mortenson (1997)). In the same way a surface, representing the change in two input variables, can be generated from 16 points, and a cubic geometric object representing three inputs can be generated from 64 points. Each of the 64 Performance Points has, in this case, to be simulated with 5 replications. Then a total of 320 runs have to be made. The use of a cubic geometric object has not yet been tested. In this article, only the use of a surface is described. The reduction of necessary runs is still large. In the five Kanban Cells case, there are three inputs; the number of cards in Cell *B*, Cell *C* and Cell *D*. To illustrate the method, the inputs representing Cell *B* and Cell *D* are used to generate one surface for each value of input Cell *C*. In this way, the method can handle three inputs by using a number of surfaces that is not a cubic object. The number of necessary simulated Performance Points becomes 16 for each value of input Cell *C*, thereby a total of 144 performance points have to be simulated with a total of 720 runs. For every value of input Cell *C* the method repeats itself. So in this article, only the performance surface related to input Cell *C* with value 1 is presented in detail.

For this case, the elements of $f(u, v, w) = (g_1(u, v, w), g_2(u, v, w), g_3(u, v, w))$ become:

Inputs: $v \in [0,1]$ represents the number of cards in Cell *B* (input Cell *B* = 1, ..., 9)
 $w \in [0,1]$ represents the number of cards in Cell *C* (input Cell *C* = 1, ..., 9)
 $u \in [0,1]$ represents the number of cards in Cell *D* (input Cell *D* = 1, ..., 12)

Responses: g_1 is the average Leadtime
 g_2 is the average Throughput
 g_3 is the average WIP

For illustration purposes, the case is simplified even more by setting the number of cards in Cell *C* to 1, as already described.

The model becomes $f(u, v) = (g_1(u, v), g_2(u, v), g_3(u, v))$

The choice of the 16 configurations to be simulated is done by first finding the extreme values of each input. In this case, Cell *B* equal 1 and 9, and Cell *D* equal 1 and 12. From these, the four corner points of the surface are generated, see table 1. Second, choose two internal values on each input. From experience we choose Cell *B* equal 2 and 7, and Cell *D* equal 2 and 5.

The accuracy of the estimated performance points is dependent on the choice of these values. Our future research will result in a detailed description of how to make this choice.

All the 16 simulated configurations are presented in table 1. It is important to note that the order of the configurations is important due to the Bezier interpolation methodology.

Configuration	C 1	C 2	C 3	C 4	C 5	C 6	C 7	C 8	C 9	C 10	C 11	C 12	C 13	C 14	C 15	C 16
Cell A	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Cell B	1	1	1	1	2	2	2	2	7	7	7	7	9	9	9	9
Cell C	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Cell D	1	2	5	12	1	2	5	12	1	2	5	12	1	2	5	12
Cell E	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Corner			Corner									Corner			Corner

Table 1. Number of cards in each cell for the simulated configurations

Based on these 16 points, in the three dimensional space, a Bezier surface of the second degree is established (cf. Farin (1993)) see figure 8. The gridded area of figure 8 is the Performance Surface of the system while the number of cards in kanban cell B and cell D are varied along the u and v vectors. A surface like this can be described and evaluated using the parametric values u and v . A point $\mathbf{p}(u_i, v_j)$ on the surface, also with other values of u and v than in the simulated configurations, has the coordinates X, Y, Z given by u and v .

$$X = g_1(u_i, v_j), Y = g_2(u_i, v_j), Z = g_3(u_i, v_j)$$

From a point on the performance surface given by u_i and v_j the corresponding performance average lead time ($X = g_1(u_i, v_j)$), average throughput ($Y = g_2(u_i, v_j)$), and average WIP ($Z = g_3(u_i, v_j)$) can be estimated by $f(u, v)$.

Figure 8. Bezier surface based on 16 Performance Points

To test the accuracy of the method, all estimated performance points of one system performance surface were simulated. Each estimated performance point was compared to the

related simulated performance point. All the distances, between an estimated and the corresponding simulated performance point, were inside the 90 % confidence interval of the simulated performance point. The choice of the internal points, to be simulated in each parameter direction, is critical in order to achieve the best possible accuracy. Future research on this topic will describe a selection methodology that ensures good accuracy.

Conclusions and Extensions

System performances presented in a 3D space give the system developer and user a more complete insight in the behaviour of their system under different configuration situations. The inclusion of measures of variation in the evaluation will secure short time (term?) performances, in contrast to only long term steady state average measures. Previous performance presentation methods do not provide this opportunity. However, the main advantage of presenting performance points in 3D environment is that it provides the opportunity to use geometrical tools to estimate complementary performance points from a limited set of simulated configurations.

A two-dimensional performance surface in a 3D space is studied in this article; varying numbers of cards in two Kanban Cells when the numbers of cards in the other Kanban Cells are held constant. In future research, an investigation where cards in more than two Kanban Cells are varied at the same time, should be carried out. This will create a multi-dimensional performance space, - a performance space more difficult to illustrate, but even more useful for testing practical applications. This would probably reduce the amount of necessary simulations even further. The choice of the internal points to be simulated in each parameter direction, and a methodology for estimations of the variance performance at the estimated performance points, will also be addressed in future research.

Acknowledgements

The authors are thankful to valuable comments from anonymous referees. This study has partly been supported by the Swedish Research Council Formas (project SE +468200440).

References

- Banks, J., Carson II. J.S., Nelson, B. L., Nicol, D. M., 2005. *Discrete-event system simulation* (4 ed). Upper Saddle River, N.J.: Prentice-Hall
- Barton, R. R., 1998. Simulation Metamodels, *Proceedings for the 1998 Winter simulation Conference*.
- Barton, R. R., 2006., Metamodel-Based Simulation Optimization, Handbook in OR&MS, vol. 13, chapter 18
- Browne, J., Harhen, J., Shivan, J., 1996. *Production Management Systems* (2 ed). Upper Saddle River, N.J.: Prentice-Hall
- Farin, G., 1993. *Curves and surfaces for computer aided geometric design: a practical guide*. Boston : Academic Press
- Fu, M. C., 2002. Optimization for Simulation: Theory vs. Practice, *INFORMS Journal on Computing*, 14, 3, 192-215
- Gstettner, S., Kuhn, H., 1996. Analysis of production control systems Kanban and CONWIP. *International journal of production research*, 34, 11, 3253-3273
- Herer, Y. T., Shalom, L., 2000. The Kanban assignment problem – A non-integral approach. *European Journal of Operational Research*, 120, 260-276

- Hopp, W. J., Spearman, M. L., 2000. *Factory Physics: Foundations of manufacturing management* (2 ed.). New York: Irwin/McGraw-Hill
- Kirkavak, N., Dincer, C., 1999. The general behaviour of pull production systems: The allocation problems. *European Journal of Operational Research*, 19, 479-494
- Köchel, P., Nieländer, U., 2002. Kanban optimisation by simulation and evolution. *Production Planning and Control*, 13, 725-734
- Lee, H-T., Wang, M. H., 2007. On the search of workstations arrangement in pull production systems. *Computers and Industrial Engineering*, 54, 3, 613-623
- Little J. D. C., 1961. A Proof of the Queuing Formula $L = \lambda \cdot W$. *Operations Research*, 9, 383-87.
- Mortenson, M. E., 1997. *Geometric Modelling*. New York: Wiley
- Muckstadt, J. A., Tayur, S. R., 1995a. A Comparison of alternative kanban control mechanisms. I. Background and structural results. *IIE Transactions*, 27, 140-150
- Muckstadt, J.A., Tayur, S. R., 1995b. A Comparison of alternative kanban control mechanisms. II. Experimental results. *IIE Transactions*, 27, 151-161
- Pettersen, J.A., Segerstedt, A., 2008. Restricted Work In Process, A study of differences between Kanban and CONWIP. *International Journal of Production Economics*, to appear
- Rolstadås, A., Andersen, B., 2000. *Enterprise Modeling*. London: Kluwer
- Savsar, M., Al-Jawini, A., 1995. Simulation analysis of just-in-time production systems. *International Journal of Production Economics*, 42, 67-78
- Silver, E. A., Pyke, D. F., Peterson, R., 1998. *Inventory Management and Production Planning and Scheduling* (3 ed). New York: Wiley
- Tayur, S. R., 1993. Structural Properties and a Heuristic for Kanban Controlled Serial Lines. *Management Science*, 39, 11, 1347-1368
- Yu, B., Popplewell, K., 1994, Metamodels in manufacturing: A review, *International Journal of Production Research*, 32, 4, 787-796