doi:10.1108/01443571211284197 • www.stefancreemers.be • info@stefancreemers.be

Extending the production dice game

Marc Lambrecht
Stefan Creemers
Robert Boute
Roel Leus

Purpose: **The production dice game is a powerful learning exercise focusing on the impact of variability and dependency on throughput and work-in-process inventory of flow lines. This paper seeks to extend the basic dice game along the following lines. First, it will allow operations to take place concurrently as opposed to sequentially, which works better in a classroom setting. Second, it will allow both starvation and blocking of the line. Third, it will consider balanced lines with workstations characterized by different degrees of variability. Finally, it aims to use different sets of dice in order to represent a wide range of variation coefficients of the production line. The obtained insights can be extended to a supply chain context as well. The developed game can be played on-line and the software is freely downloadable.**

Design/methodology/approach: **The paper extends the dice game and offers an easy-to-use simulation tool.**

Findings: **The key aspect of students' learning experience is the understanding of the relationship between variability and throughput in an environment with dependent workstations and limited buffers.**

Originality/value: **A rather complicated research question is transformed into an easy-to-use simulation tool that in no time can be used by practitioners and students.**

Keywords - **dice game, variability, throughput, simulation, buffers, indoor games, learning, students**

1 Introduction

The production dice game is a learning exercise which demonstrates the impact of variability and dependency on throughput and work-in-process inventory. The game deals with flow-shop layouts, i.e. layouts in which equipment or work stations are arranged according to the progressive steps in which a product is made. A good example is an assembly line, in which the path of a product is a straight line. Unfortunately, the production rate of work stations can be highly variable due to all sorts of outages (e.g. machine failures, repairs, minor stoppages, changeovers). Variability is inherent in almost every production environment. The work stations are therefore usually buffered with inventory, which means that work-in-process is stored in front of each station. These buffers can serve to absorb (part of) the variability in the line, but the buffer space between two stations is usually limited. As material moves from one station to the next along the line structure, dependencies are created, i.e. certain operations cannot begin until other operations have been completed. We provide an overview of the literature on the dice game in section 2.

The combination of dependency and variability creates machine interactions, which take the form of either starvation or blocking and this ultimately has an impact on the throughput (output per unit time) and the level of work-in-process of the flow-shop. Consider two consecutive machines. If the upstream machine fails to produce, the downstream machine may become starved because its input buffer is empty and therefore is forced to be idle. If the downstream machine fails, the upstream machine may become inactive because it is blocked due to the limited buffer space between the two machines (the buffer fills up). The frequency of starvation and blocking (and consequently the amount of idle time and lost throughput) depends on the size of the buffers. Buffers defer idleness and consequently increase throughput, but of course at the cost of increased inventory.

The dice game can be used to illustrate these concepts. It proceeds as follows. Each player in the game represents a work station, so n players represent a sequential line of n work stations. At each step in the game, the production output of each work station is determined by rolling the dice. In this way, we introduce variability in the output of the station. Inventory buffers are positioned between adjacent work stations. Large buffers are able to absorb more variability and thus allow machines to operate more independently. Smaller buffers, on the other hand, will create more starvation and blocking but are less costly in terms of in-process inventory and cycle times.

The dice game can be played manually (we briefly discuss the manual game in section 3) using dice (to determine machine output) and coins or chips (the parts or products that move through the production line), but it can also be implemented as a simulation-based computer application. Our implementation of the game can be played on-line and the software is also freely downloadable. The game can be accessed at the following website: http://www.econ.kuleuven.be/Dicegame. A typical layout is displayed in Figure 1. Its behaviour conveniently matches that of a real production line. The insights obtained from the game can therefore be transferred immediately to real-life situations. It is interesting to note that the dice game can also be used in a broader supply chain context. Each work station can be interpreted as a unique organization (customer, distributor, production plant,) characterised by various sources of variability such as machine breakdowns, material shortages, quality errors, bottlenecks, demand variability and replenishment rules. Inventory is maintained throughout the supply chain to buffer against this variability. Several well-known supply chain practices (e.g. promoting information visibility, managing the bullwhip effect, improving process control etc) to reduce variability can then be discussed in class. The game can even be used in non-manufacturing environments. The dice game can be used in a service operations class. Think e.g. of healthcare operations which are also very much characterised by variability and flow constraints. In our paper we take a product-based view but the flow units can easily be interpreted as customers or patients. Consequently, the dice game could be used by lecturers of different courses - an engineering operations management module, a service operations module or a healthcare module. The game's setup is limited and straightforward, which makes it very attractive to play in a classroom setting. The computerised online version builds on this and is particularly useful for subsequent iterations of the game being played by students outside the classroom.

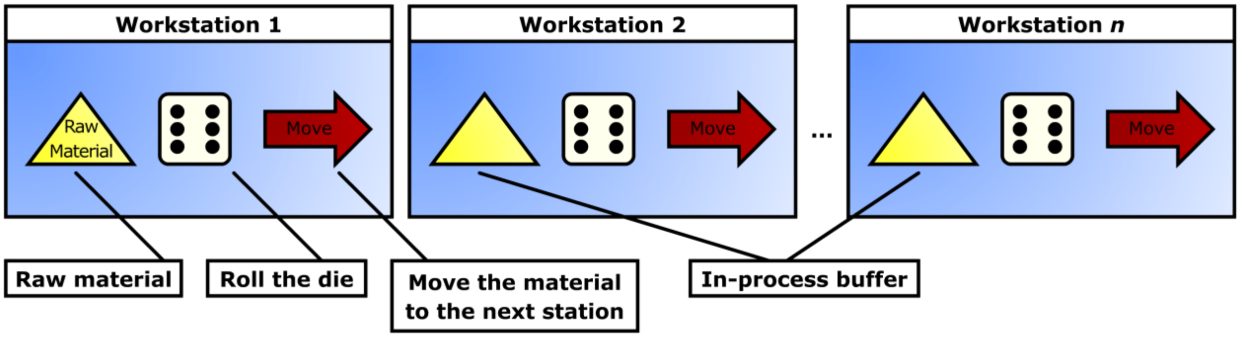

Figure 1: The basic flow-shop layout of the dice game

2 Literature review and overall game setup

The dice game has been extensively described in the literature, resulting in a wide range of game variants. The concept of the game can be traced back to Goldratt's "boy-scout hike" (Goldratt,1992). It was Alarcón and Ashley (1999) who then related the dice game to lean production concepts and used project management as an example. Hilmola (2006) in turn used the dice game to introduce system dynamics. The papers by Umble and Umble (2005) and Johnson and Drougas (2002) describe the classroom use of the dice game in combination with spreadsheet simulation. Francis (2009) also uses both spreadsheet simulations and a manual version of the dice game to illustrate the importance of quantitative techniques in operations management. In Umble and Umble (2005) both balanced and unbalanced lines are considered, while Johnson and Drougas (2002) focuses more on the technicalities of spreadsheet modelling. These last two papers constitute the basis for the model presented in this article. In the following paragraphs, we outline how exactly our paper extends the basic dice game.

First, we allow that the operations take place concurrently as opposed to sequentially, which is the more traditional way of playing the game. In a traditional sequential setup the work stations operate sequentially within a given time period, which implies that the processed units of a work station are available to the subsequent work station within a single period in the game (the second person rolls the dice after the first person has rolled the dice). This is not the case when the experiment is conducted concurrently: in this case every work station operates simultaneously within a given time period, independent of the production of the upstream and downstream work stations. This means that the production output is only based on the results of rolling the dice and the surrounding buffers at the beginning of the period. The reason why we propose a concurrent variant of the game in this article is because this setting better reflects paced production lines. Within a given "takt time" (defined in this paper as a period in the game) all work stations perform their work simultaneously and the goods in process move to the next station at the start of a new cycle (takt time). As stated in Johnson and Drougas (2002), the steady-state results achieved using either approach (concurrent and sequential) will be comparable although the probabilities of starvation are different). In literature, both the sequential and concurrent versions are discussed. We opt for the concurrent version because it works better in a classroom setting.

Second, we allow both starvation and blocking of the line, whereas in most papers only starvation is considered. In many realistic automated production settings the storage areas can hold only a finite amount of material. It is because of this limited buffer capacity that blocking can occur. When the maximum buffer size is reached, the upstream process is blocked. In the simulation we therefore impose an upper limit on the in-process inventory at each work station.

Third, we use different sets of dice in order to represent a wide range of variation coefficients of the production rate (Umble and Umble (2005) also allows for different levels of variability). This will be further explained in the next section, where our model is described in detail.

Finally, we deal with balanced lines (the expected production output is the same for every work station), but the work stations under consideration can have different levels of variability. It is, for example, interesting to examine what the impact on system performance is of a high-variability work station at the start of the production line compared to a high-variability station at the end of the line. This extension is certainly valid in a supply chain context, where variability can be created in different stages of a supply chain (e.g. during transportation, in a manufacturing facility or a distribution centre).

Thanks to these four extensions, our dice game becomes a unique approach which reflects real-life characteristics more accurately. Although we do recommend instructors who are interested in playing the dice game to be very well informed about the theoretical background of the game, it is beyond the scope of this paper to discuss the theoretical insights obtained regarding blocking and starvation in buffered serial lines. We refer to the following papers for an excellent overview of the theoretical work on the subject: Baker, Powell and Pike (1990); Conway, Maxwell, McClain and Thomas (1988); Dallery and Gershwin (1992).

3 The simulation tool

In this section we describe our simulation tool for the dice game. The tool is available as a stand-alone executable file that requires no specific additional pre-installed software to run. The Dice Game software was implemented in Macromedia Flash using the ActionScript programming language. ActionScript is a cross-platform, object-oriented scripting language that allows the creation of stand-alone and web-based applications. The game can be played at the following website: http://www.econ.kuleuven.be/Dicegame.

The tool's main interface is shown in Figure 2. If the user clicks on the question-mark button on the screen, a general help file will be displayed, whereas mousing over the button displays screen-specific help.

The serial production line consists of five work stations (represented by circles). The first work station can be considered as a receiving area of raw material, which means this station can never be starved. The four other stations are processing units. The last work station delivers the finished product and can therefore never be blocked. There are two rectangles at the upper left side of the circles: the first rectangle indicates the amount of inventory at the start of the game (the starting inventory) and the second contains the maximum number of units allowed in the buffer (the buffer size). These two values function as input to a simulation run and consequently do not change during the game. The square in the middle of the circle displays the result of rolling the dice (the dice on the upper left side shows the outcome of the previous role of the dice). It is these results that determine the production output of the work station.

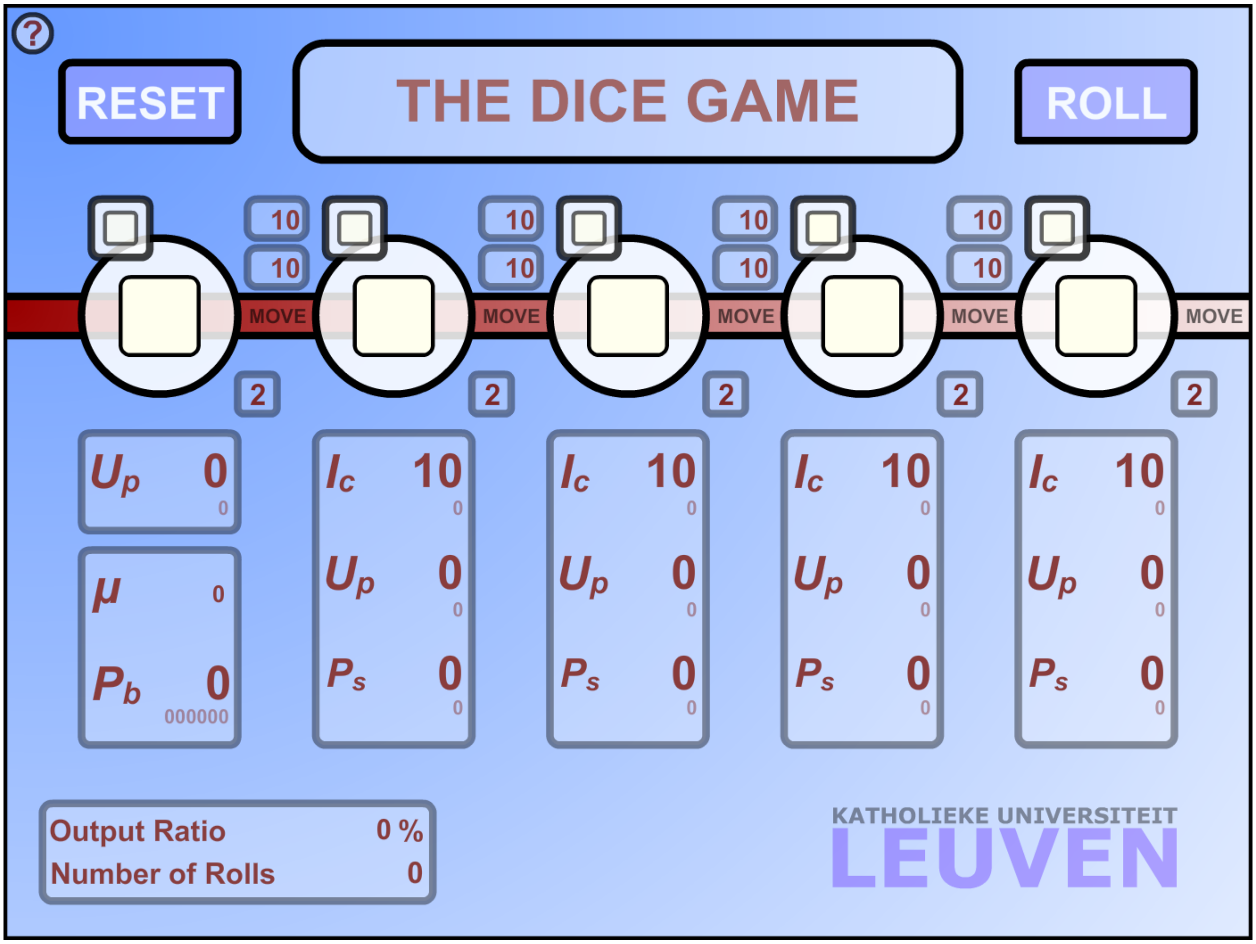

Figure 2: The program interface

The type of dice used is indicated in the box at the lower right side of the circle. We propose six types of dice, as shown in Figure 3, which allow for different levels of variability, ranging from level 1 up to level 6. The average value of the dice is always 3.5 but the squared coefficient of variation ranges from 0.02 (level 1) up to 3.082 (level 6). Three of the dice types have six sides and the other three have 20 sides. Note, however, that the sides do not necessarily display each value between one and 20, in order to correspond with the correct variation level and average. The player determines which types of dice will be used and thus controls the variability in the production output of a station. One can choose the same dice for all work stations or different dice at different work stations. This allows the simulation of production lines with the same level of variability for all stations or production lines with high/low variability at the start/middle/end of the line. Each roll of the dice represents the potential production output of a work station during one period. This step can be repeated anywhere from 50 up to 5000 times (periods).

Figure 4 depicts the input screen. This part of the game is very easy. The player determines (1) the type of dice per work station; (2) the starting inventory at each work station; and (3) the maximum buffer content. The player can input and change the abovementioned parameters by clicking in the appropriate cells (we refer to the numbers in blue on the screen) and next inputting the numbers. After clicking on the button SUBMIT and selecting the number of dice rolls (button ROLL), the player can either activate the simulation of a single period in the game, or choose to generate multiple periods (between 50 and 5000) at once.

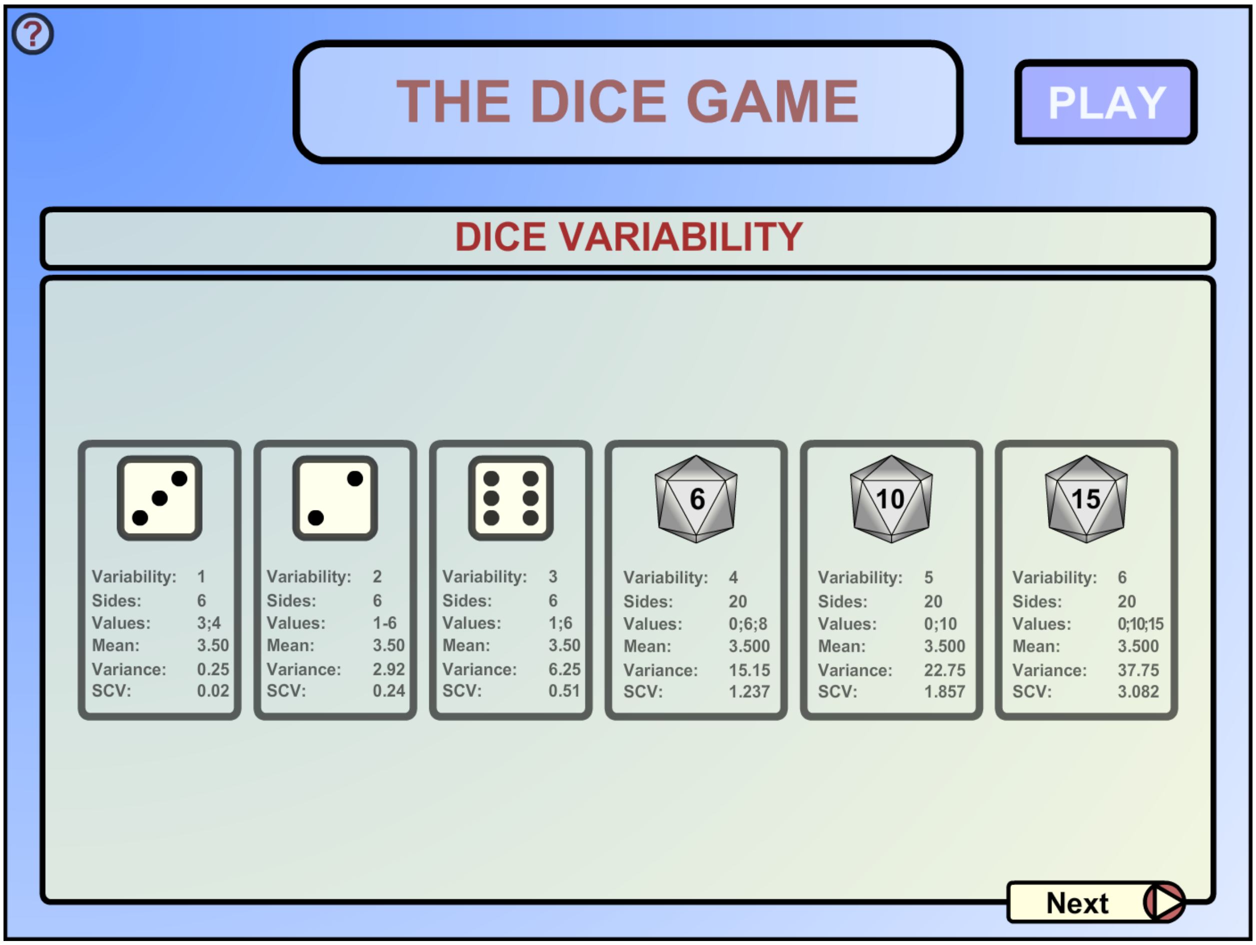

Figure 3: The different types of dice

We advise players to set the buffer size so that it either exceeds or equals the maximum value of the dice selected (in order to simulate the correct average value). It is also advisable to set initial inventory to exceed or equal the maximum buffer size (in this way the steady-state condition will be reached faster).

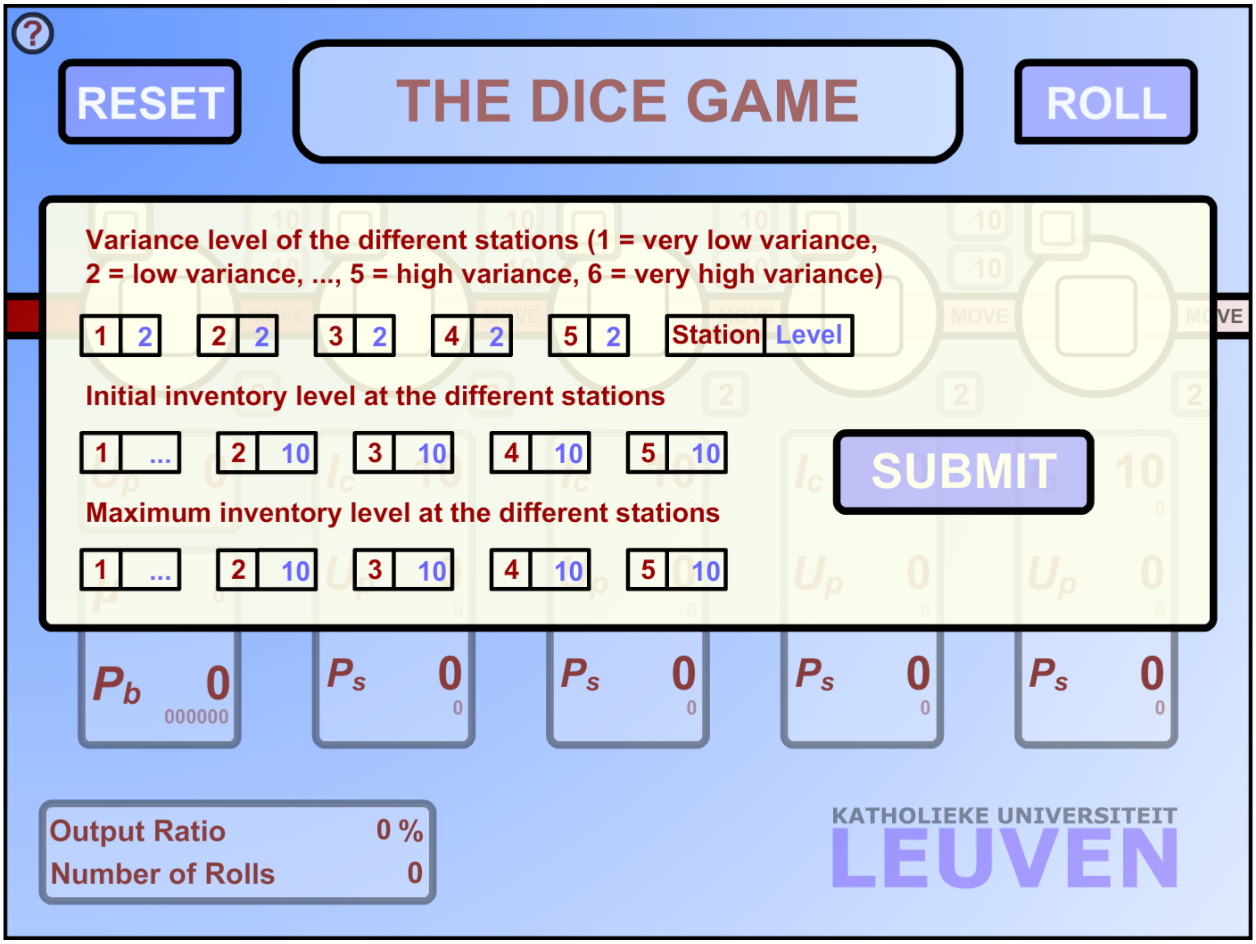

Figure 4: The input screen

The dice establishes the potential output of the work stations. The actual output, however, may differ because of starvation and blocking. We illustrate this important aspect of the game with the example shown in Figure 5. Each station has an input buffer and an output buffer; the latter is actually the input buffer of the next station. We refer to this input buffer as the inventory of the work station. In our example both buffers have an upper limit of eight units. The current inventory of the work station analysed here equals four, while the current inventory at the next station equals five. The outcome of rolling the dice is six. Let us have a closer look at the actions which are then triggered.

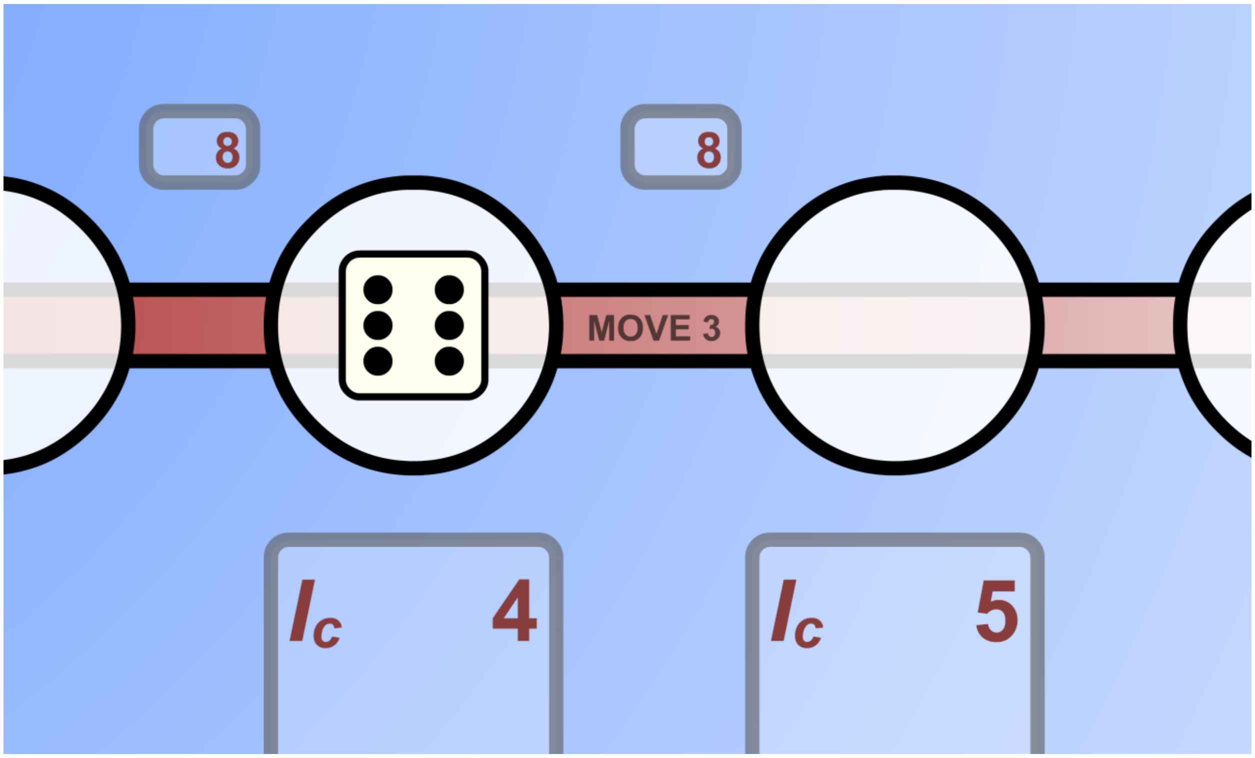

Figure 5: Potential versus actual output

Rolling the dice yields a potential production quantity of six, but since there are only four units available in the input buffer, two units are starved. Moving the four units to the buffer of the next station would result in an inventory of 9 (5+4), which is more than the maximum buffer size (which was set at 8). Therefore, the number of units that will actually move from this work station to the next (i.e. the actual output) is three: we lose two units because of starvation and one because of blocking.

These computations are performed for each work station at the beginning of each period. Once the actual output for each work station has been determined, the actual production output is transferred to the next station (except for the output of the last station, which is the finished product). All work stations operate simultaneously (i.e. concurrently). This is an important aspect of the dice game. Before each run (period), the availability of work-in-process is checked, and it is not permitted to use material that is not available at that point in time. In a sequential version of the game, the preceding operation's processed units are available to the subsequent operation within the same period.

We refer to Figure 6 for a discussion of the output of the game. Here we select type 2 die, a starting inventory of ten units at each work station and a buffer limit of ten units. The system is run for 1000 dice rolls.

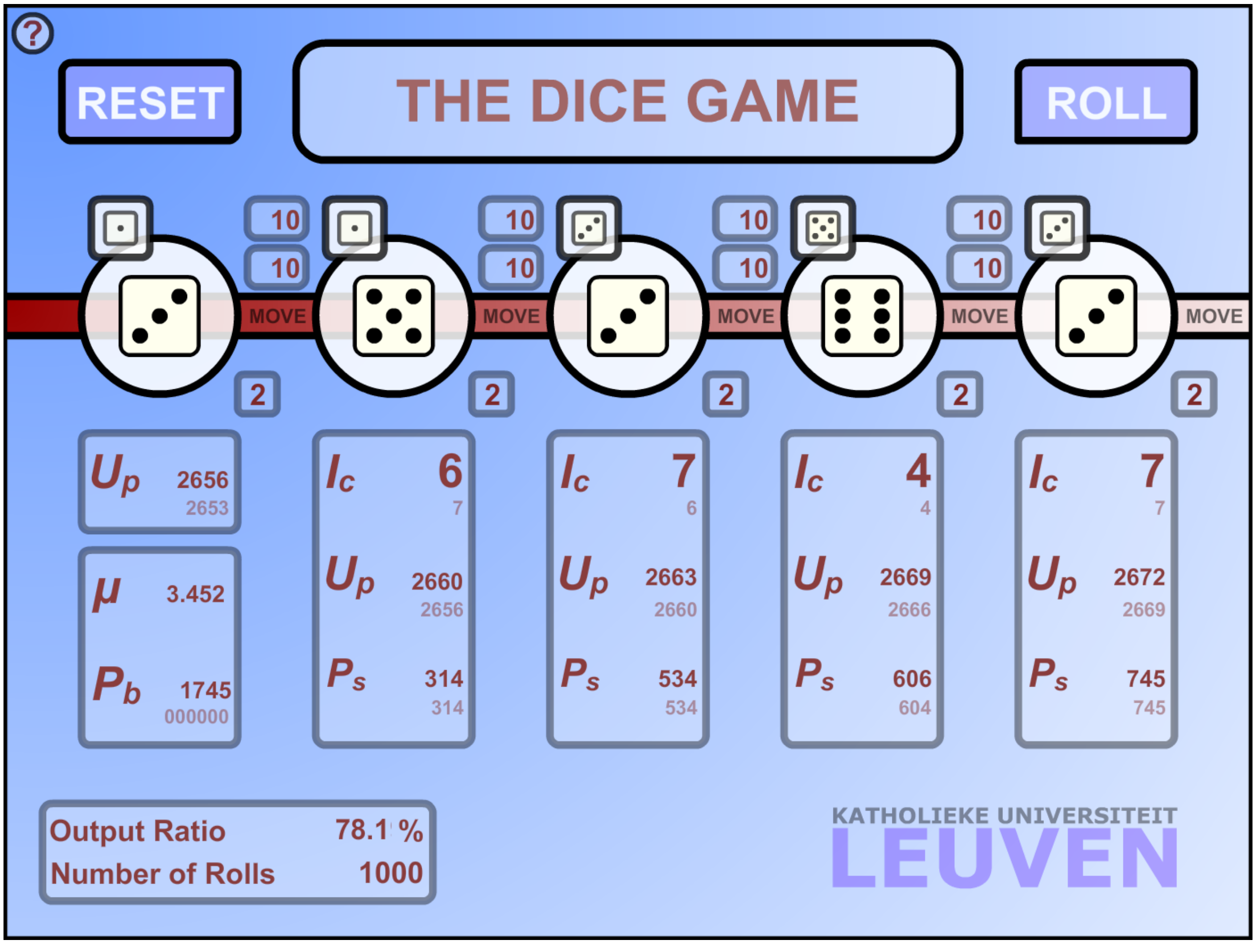

Figure 6: Output of a simulation run

In Figure 6 the following symbols are used. P_b stands for the total number of units blocked across all stations. P_s represents the number of units starved for a particular work station. U_p equals the actual output (units moved to the next station) and I_c indicates the current inventory (input buffer). Finally, μ stands for the average result of all dice rolls (grand average).

If we play the game 1000 times, we have a potential output of 3500 units (1000 3.5). However, due to starvation and blocking the actual output will be substantially lower. Suppose, for example, that the actual output (number of units moved (completed) at the last work station) is 2672 and that the number of units starved at the last station equals 745 units. The total "demand" at the last station equals 2672+745= 3417 (while the expected value is 3500 units). The output ratio consequently equals (2672/3417) = 0.781 or 78.1%. This is our metric for measuring throughput. The difference between 3417 and 2672 is the output shortfall.

As mentioned in the introduction, the dice game can be played as a computer-based simulation or as a hands-on manual game. A useful layout for the manual game is presented in Figure 1 (while others may prefer a layout as provided in Figure 2). The rules of the game are exactly the same as those explained above. Each player is in charge of one work station. In step one, the players roll the dice. In step two, the player has to determine how many units will be transferred to the next station. That, of course, depends on the number indicated on the die, the available inventory in the input buffer and the maximum inventory allowed in the buffer of the next station (this step is illustrated in Figure 5). In a third step all players move simultaneously the units (chips) to the next station. The game can be played with different sets of dice reflecting the level of variability. During the game, information can be gathered and summarized in a table such as the one shown in Table 1. Table 1 focuses on work station 5. It is not necessary to collect data for the other work stations.

	Outcome of the die	Finished products sent to the customer	Number of units starved
Period 1 Period 2 ...			
Sum		U_p	P_s
Output ratio	$\frac{U_p}{U_p+P_s}$		

Table 1: Data sheet for the manual game

The manual game will familiarize the students with the dice game and thus facilitate the transition to the computerized version of the game. We do advise instructors to play the game for at least 20 periods and make clear to the students that the steady-state conditions will not be reached. We advise to run the game in 2 or 3 rounds enabling the teacher to adapt the system to show the effect of adding (reducing) buffer inventory and/or restricting variability. After playing the manual version of the game, the teacher has to spend some time for debriefing the class on the results of the game. Ask questions such as : What did you observe? What's causing the throughput loss? What options do you have to increase the throughput? It's worthwhile to point out to the students that these observations do not only hold for flow lines, but are also valid in more general supply chain settings. Next you move to the computer-based simulation. This offers the opportunity to experiment with long runs resulting in statistically significant results.

4 Insights from the dice game

The key aspect of students' learning experience is the understanding of the relationship between variability and throughput in an environment with dependent work stations and limited buffers. Limited buffers create starvation and blocking and this in turn impacts on throughput. Although low-inventory environments are typically discussed and advocated in courses dealing with lean manufacturing, the dice game clearly illustrates the prerequisite of reducing variability in achieving lean operations. The game illustrates that in the presence of variability, so-called lean practices have a negative impact on throughput. In that case, adding buffers is the only way to improve throughput performance. The amount of “productive” buffer capacity depends on the degree of variability. There are many process improvement tools focusing on variability reduction (Total Productive Maintenance, Six Sigma,). These tools will improve throughput rates and reduce the need to hold large buffers. What is required is a balanced view of dealing with variability: a balance between holding buffer inventory or reduction of variability itself. We recall once more that these insights not only hold for flow lines, but for supply chain settings in general. A very simple way for students to gain these important insights is by playing the dice game, as it requires only a 15-minute introduction.

After playing the game manually, the simulation tool can be used. We advise the instructor to start with a production line whose output potential is determined by type 2 dice. These dice are standard six-sided dice (the number of spots ranges from 1 to 6). We limit buffer capacity to 10 and initial inventory is also set at 10. The students are then asked to guess what the output ratio will be. Next, a simulation of 1000 dice rolls is performed, which will yield an output ratio of approximately 78% (slightly varying from experiment to experiment). In other words, the throughput loss amounts to more than 20% and students will typically underestimate this value considerably. The experiment is then repeated with type 1 dice (six sided but only with 3 or 4 spots on each side), and with the same buffer sizes. The output ratio will now be approximately 97.5%, which constitutes an impressive improvement. This simple experiment usually stimulates students to examine this in greater detail.

At this point the instructor can introduce more experiments, preferably in small groups. Each group is assigned a type of dice and evaluates the output ratio for different buffer sizes. The instructor can then collect the data and summarise the results, yielding a graph similar to Figure 7 (the y-axis refers to the output ratio, the x-axis refers to the buffer capacity and the different curves refer to the different levels of variability characterised by their squared coefficient of variation as given in Figure 3).

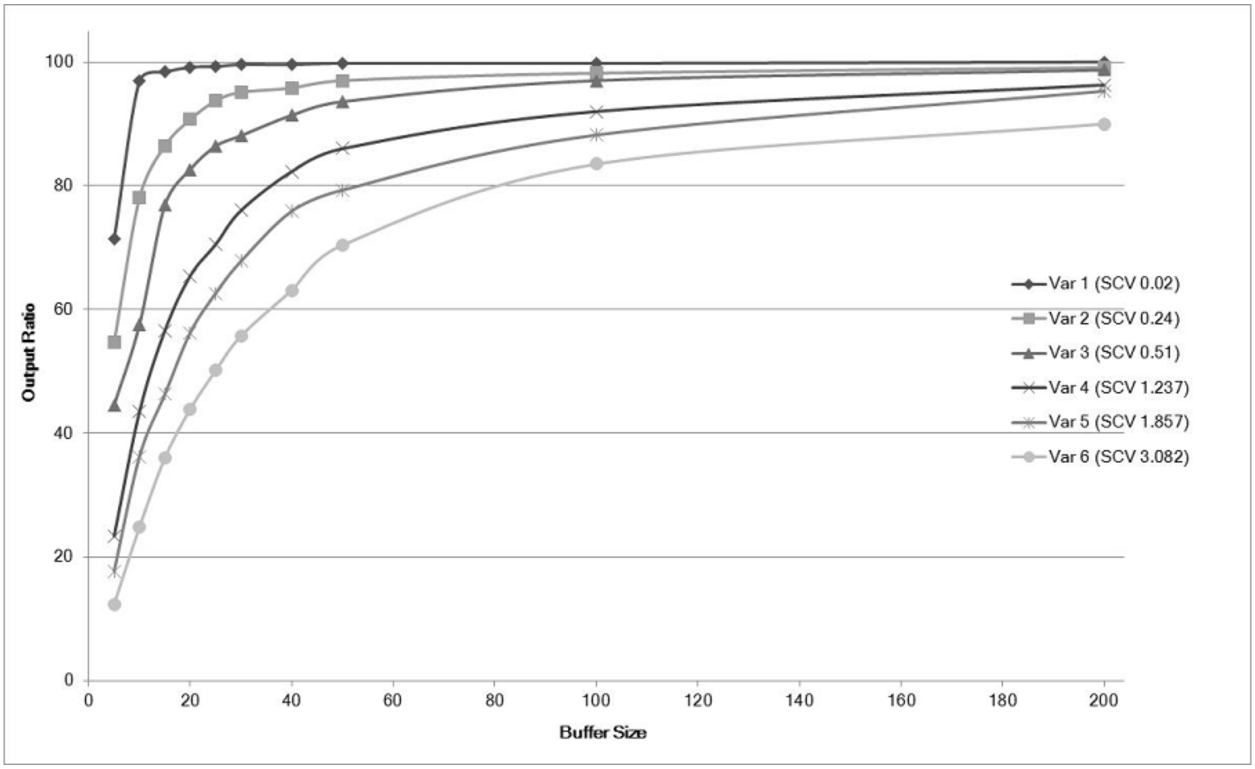

Figure 7: The concave curve relating buffer size to output ratio

Figure 7 illustrates the relationship between buffer size (the same maximum buffer size for all work stations) and the output ratio for the different types of dice. For a buffer size of 20, for example, the output ratio ranges from 42% to 99%. At this point during the class sessions, it is advisable to introduce some theoretical insights explaining the simulation results. The paper by Conway et al. (1988) can be very helpful in that respect. Based on Figure 7 and Conway et al. (1988) we can formulate the following insights. Note that these insights hold for balanced, equally buffered lines.

Insight 1 *The output ratio/buffer size curve is sharply concave. The more variability, the higher the throughput loss.*

Insight 2 *To achieve a given output ratio, the buffer capacity should be proportional to the squared variation coefficient of the processing rate.*

Insight 3 *A small amount of work-in-process is very helpful, but its usefulness gradually decreases. The rate at which it decreases depends on the level of variability.*

Burman, Gershwin and Suyematsu (1998) illustrate how helpful the insights above can be to improve the design of a printer production line. In Hopp and Spearman (2000), the reader will find an excellent discussion of the corrupting influence of variability.

After the experiments the instructor can ask students what the major sources of variability are, and what can be done from an engineering point of view to eliminate these sources of variability. This approach allows the game to be positioned in a broader context.

Another interesting experiment is to investigate what happens without an upper limit on the buffer size. As blocking will be impossible, the only possible form of interference in this case is starvation. We can simulate such a situation by setting the maximum inventory level at 999 (see Figure 4). For the parameters shown in Figure 6, but now with unlimited buffers, we obtain an output ratio of approximately 98%, in contrast to the 78% mentioned earlier when both blocking and starvation were possible. These findings lead us to Insight 4.

Insight 4 *In designing production lines, it is of crucial importance to accurately assess the in-process inventory space that is required. Limiting the buffer space may result in a dramatic output shortfall.*

We refer to Gershwin (1987) for an excellent theoretical treatment of this problem.

A final set of experiments that we recommend to perform in class is related to balanced lines (lines with the same expected output for all stations) with unequal variability. So far we have analyzed production lines with equal output potential, both in terms of expected output and variability (by using the same type of dice for all work stations). In reality, it is very well possible that the variability of the output is different across the various work stations. This can easily be simulated by means of the dice game: we simply select a different type of dice for each work station. The following questions might be interesting to examine in more detail. Does this have an impact on throughput performance? Do we have to increase the buffer capacity to attain a particular output ratio? In which positions is extra buffer space most effective?

One possible experiment proceeds as follows. We start with the standard situation described above, i.e. type 2 dice and a buffer capacity of 10 units. The output ratio of this setting was said to be typically 78%. Changing the type of dice for work station 3 to type 5, while leaving the buffer capacities unaltered, causes the output ratio to drop to 55%. To re-attain the original target of 78%, the following steps can be taken. First, increase the input buffer capacity of work station 3 to 20 units; no significant improvement will be observed. Next, increase the output buffer capacity of work station 3 to 20 units as well. This will result in an output ratio of 74%, which indicates how beneficial the strategy of increasing both the input and the output buffers can be. Further improvements can be made by adding extra buffer space to stations 1 and 5. Take, for example, 15 units and the output ratio will increase to 78%. This suggests that it is not sufficient to protect the high-variability work station alone. We also have to protect the other work stations, because the variability propagates over the whole line. In other words, a bell-shaped allocation of buffer capacity (centred around the high-variability work station) is advisable see (Conway et al. 1988).

Insight 5 *Work stations with large variability have an impact on the output performance of the whole line. This means that a high-variability work station has to be protected by extra buffer capacity. Because of the propagation of variability the other work stations need extra protection as well. A bell-shaped allocation is therefore advisable.*

5 Conclusion

The dice game is a powerful learning exercise focusing on the impact of variability and dependency on throughput and work-in-process inventory of flow lines. In this paper we take a product-based view, but the game can be used in a service as well as a manufacturing setting. In this paper we offer both a manual and a simulation-based tool to play the game. The insights obtained from the game can help to optimise the design of production lines. Management and engineering students should pay attention to the size of the in-process inventory space. The availability of buffers prevents blocking or starvation of other work stations due to the variability of each machine's production. All this indicates that variability reduction itself is a key focal issue for management. Although this does not come as a surprise for most students, our experience indicates that students, as well as managers, usually underestimate the impact of variability.

References

[1] Alarcón, L. and Ashley, D. (1999), "Playing games: evaluating the impact of lean production strategies on project cost and schedule", in Proceedings of the 7th Annual Conference of the International Group for Lean Construction at University of California Berkeley, CA, pp. 263–274.

[2] Baker, K.R., Powell, S.J. and Pike, D.F. (1990), "Buffered and unbuffered assembly systems with variable processing times", *Journal of Manufacturing and Operations Management*, Vol. 3, pp. 200–223.

[3] Burman, M., Gershwin, S.B. and Suyematsu, C. (1998), "Hewlett-Packard uses operations research to improve the design of a printer production line", *Interfaces*, Vol. 28, No. 1, pp. 24-36 .

[4] Conway, R., Maxwell, W., McClain, J.O. and Thomas, L.J. (1988), "The role of work-in-process inventory in serial production lines", *Operations Research*, Vol. 36, No. 3, pp. 229–241.

[5] Dallery, Y. and Gershwin, S.B. (1992), "Manufacturing flow line systems: a review of models and analytical results", *Queuing Systems*, Vol. 12, pp. 3–94.

[6] Francis, V.E. (2009), "Winning hearts and minds: an argument for quantitative analysis in an operations management course", *Decision Sciences Journal of Innovative Education*, Vol. 7, No. 1, pp. 73–79.

[7] Gershwin, S.B. (1987), "An efficient decomposition method for the approximate evaluation of tandem queues with finite storage space and blocking", *Operations Research*, Vol. 35, No. 2, pp. 291–305.

[8] Goldratt, E. and Cox, J. (1992), *The Goal: A process of Ongoing Improvement*, 2nd Revised Edition, Great Barrington, Massachusetts, North River Press.

[9] Hilmola, O.H. (2006), "Using Goldratt's dice-game to introduce system dynamics models and simulation analysis", *International Journal of Information and Operations Management Education*, Vol. 4, No. 4, pp. 363–376.

[10] Hopp, W.J. and Spearman, M.L. (2000), *Factory Physics: Foundations of Manufacturing Management*, McGraw-Hill Higher Education, New York, N.Y.

[11] Johnson, C. and Drougas, A.M. (2002), "Using Goldratt's game to introduce simulation in the introductory operations management course", *Informs Transactions on Education*, Vol. 3, No. 1, pp. 20–33.

[12] Umble, E.J. and Umble, M. (2005), "The production dice game: an active learning classroom exercise and spreadsheet simulation", *Operations Management Education Review*, Vol. 1, pp. 105–122.