NOTES AND INSIGHTS

Back to basics: fundamental principles of system dynamics and queueing theory

Paulo Gonçalves*

Syst. Dyn. Rev. **38**, 81–92 (2022)

Introduction

Ghaffarzadegan and Larson (2018) contend that "many traditional operations research (OR) models can be improved by including feedback processes, as is commonly done in system dynamics (SD) p. 327" and that modelers can build better models by combining the strengths of both modeling schools. Focusing mainly on queueing theory as a fundamental building block of Operations Research, they provide compelling examples supporting their claim. Similarly, Lane (1994) argued that a dialog between soft operations research and system dynamics would be mutually beneficial. In this note, I argue that models combining principles of system dynamics and queueing theory do not even have to include feedback processes to generate useful insights. To achieve this, I distill six principles, associated with the understanding of stocks-and-flows and queueing, that can help managers (and modelers) generate important insights in real settings. The principles may help educators, managers, and researchers' understanding of the dynamics of stock-and-flow structures and their implications in practice. For each principle, the note describes practical examples, connecting with previous system dynamics literature and increasing the potential for their use in teaching and modeling.

Experimental research in system dynamics shows that people struggle to understand the behavior of even simple stock-and-flow systems (Cronin *et al.*, 2009; Sterman, 2010; Sweeney and Sterman, 2000). The aspiration for this note is that the principles and associated examples can broaden the cases of simple, insightful, easy to communicate, and impactful stock-and-flow models that may help people understand the dynamic behavior of simple systems.

Fundamental building blocks for queueing and system dynamics

Before considering how to appropriately combine principles of system dynamics and queueing, we recognize important differences. Queueing emerged from the need to understand telephone traffic congestion in the

Humanitarian Operations, Università della Svizzera italiana, Lugano, Switzerland

* Correspondence to: Paulo Gonçalves, Università della Svizzera italiana, Humanitarian Operations, Via Buffi 6, Office 112, 6900 Lugano, Switzerland. E-mail: paulo.goncalves@usi.ch

Accepted by Andreas Größler, Received 19 April 2021; Revised 31 August 2021; 15 September 2021 and 23 November 2021; Accepted 26 November 2021

System Dynamics Review
System Dynamics Review vol 38, No 1 (January/March 2022): 81–92
Published online in Wiley Online Library
(wileyonlinelibrary.com) DOI: 10.1002/sdr.1699

early 1900s. The theory soon expanded to study other processes and understand the behavior of systems where random arrivals were processed by a server (Shortle *et al*., 2018). Queueing theory is interested in predicting queue length and waiting time in a queue. Applications range from telecommunication processes to traffic engineering and industrial engineering to hospital and service management. While queueing typically focuses on the long-run behavior of stationary (i.e. steady state) systems, system dynamics focuses on the long-run behavior of transient systems. For both queueing and system dynamics, stocks and flows represent fundamental building blocks. However, while stocks are treated similarly in queueing and system dynamics, flows are treated differently. In queueing, flows (or rates) are typically assumed fixed but probabilistic, while in system dynamics flows are typically variable, but deterministic. Stochastic rates are critical to queuing because sophisticated queue dynamics can only emerge in stationary queueing systems, with input rates (λ) smaller than service rates (μ)[i], when the rates are stochastic. In system dynamics, flows are typically deterministic, capturing the expected values of rates that are random in practice. Interesting queue dynamics can emerge in deterministic system dynamics models due to feedback and delays (Barlas and Özgün, 2018), i.e. dynamic complexity.

Little's Law, a cornerstone result in queuing, dictates the relationship between a Stock (L) and the flow (e.g. inflow and outflow) passing through a stationary system. When in steady state, Little's Law establishes that the long term average value of the Stock (L) is equal to the product of the long-term average arrival rate (λ) to the system and the average residence time, or average wait (w) of items in the stock. That is, the value of the Stock (L) is proportional to the arrival rate (λ). Little's Law can also be applied to system dynamics to infer the instantaneous relationship between the transient value of a Stock (L) and the instantaneous arrival rate (λ). Figure 1 shows the stock-and-flow formulations in system dynamics and queueing theory.

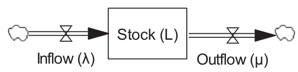
Below, I provide six examples from a real-world research and consultancy project, with useful insights that can be derived when modelers combine knowledge of system dynamics and queueing. The first two examples do not include feedback, the remaining four do. From these examples, I distill six principles, associated with the understanding of stocks-and-flows and queueing that can help managers generate insights in different contexts.

The examples draw from a collaboration with Henk Akkermans and joint work at Etel, an incumbent mid-sized telecommunications provider. At the time of our engagement, Etel was upgrading its existing copper landline infrastructure to fiber, which enabled it to offer voice-over internet protocol (VOIP) services to its installed customer base. Part of our work focused on Etel's back-office operations: as customers contracted the new VOIP services,

[i]To ensure the mathematical tractability of queueing models, they assume fixed input rates (λ) and service rates (μ). Because queueing models focus on the long-run behavior of stationary systems, it requires those fixed input rates (λ) to be smaller than processing rates (μ), (i.e. $\lambda < \mu$), to ensure that the queue does not become infinite.

DOI: 10.1002/sdr

Fig. 1. Stock-and-flow formulations in System Dynamics and Queueing

System Dynamics formulation	Queueing formulation
	*Inflow* (λ)= *Outflow* (μ) *Average wait* (w) = *Stock* (L) / *Inflow* (λ)

Etel would process and activate them. Because managers at Etel expected automated systems to handle the activation work (100% quality), they allocated minimal back-office operations capacity (Figure 2a). However, some of the contracts had problems and would literally fall out of processing, requiring rework before they were ready for activation (Figure 2b). Figure 2 captures the basic stock-and-flow structure of these back-office operations (adapted from Akkermans, 2018; Akkermans *et al.*, 2016).

Inflows and outflows are rarely in steady state

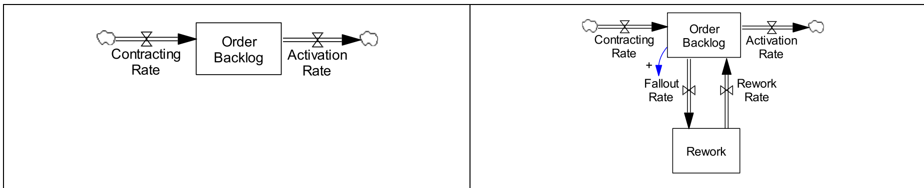
Queueing theory assumes that inflows and outflows are in steady state, with constant values (and frequently equal to each other in long-run analyses). In practice, this is rarely the case. While those assumptions allow mathematical tractability, transient mismatches between inflows and outflows are common place in practice and can lead to interesting dynamics and managerial challenges that modelers can capture and derive insights.

For Etel, the mismatch between the inflow of order fallout (i.e. problems with back-office processing of incoming customer orders) and the outflow of order rework (i.e. corrections to those faulty orders) helped managers determine the amount of resources (i.e. capacity) required to allocate to rework. In practice, managers would have regular meetings where they would compare the list of faulty orders with those that had been resolved. Still, service-capacity adjustments were made mostly in reactive mode: hiring (or firing) people based on the perceived amount of required work. In the first 20 weeks of operations, the high volume of order fallouts helped managers justify their decision to increase the number of staff dedicated to rework. Afterwards, the low volume of fallouts justified their decision to significantly reduce them. The transient nature of inflows and outflows provided an estimate for human resources requirements and guided managers' hiring and firing decisions. Figure 3a shows the data associated with these inflows and outflows. Because the volume of order fallouts reduced significantly over time, managers felt that they were able to address the root causes of the problem.

Principle:
Proper understanding of the transient nature of stocks and flows allows managers to identify important dynamic behaviors and inform specific organizational requirements. Failure to comprehend that inflows and outflows are rarely in steady state may lead to inadequate organizational decisions.

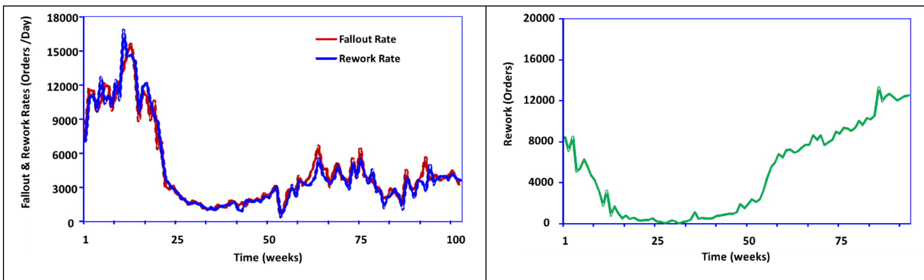
DOI: 10.1002/sdr

Fig. 2. Basic stock-and-flow structure for Etel's back-office operations [Color figure can be viewed at wileyonlinelibrary.com]



(a) Expected contract activation process | *(b) Actual contract activation process with rework*

Fig. 3. Etel's behavior over time for (a) fallout and rework rates and (b) stock of rework [Color figure can be viewed at wileyonlinelibrary.com]



(a) Fallout and rework rates | *(b) Stock of rework*

Stocks integrate the imbalance between inflows and outflows

A typical insight from elementary calculus (and captured aptly in system dynamics and queueing) is that stocks accumulate the imbalance between inflows and outflows, generating disequilibrium dynamics. Etel captured flow data associated with its processes systematically, but it did not readily capture data on major stocks. When asked for data on the stock of rework, it took managers, assisted by a team of consultants, several weeks to collect it. Because Etel's managers did not track or monitor data on the stock of rework, they were oblivious to the steady buildup of rework after week 40 (Figure 3b).

More important, managers completely missed the very high and persistent rework levels – well beyond the initial levels, when fallouts were perceived as problematic. Consequently, when Etel started struggling with high call and complaint volumes from disgruntled customers, managers were both surprised and puzzled.

Principle:
Stocks accumulate the imbalance between inflows and outflows, generating disequilibrium dynamics. In both queueing and system dynamics, stocks provide an important source of information for policy decisions. Queue length or the number of items in the stock influence important managerial decisions and should be taken into consideration. Failing to account for such important

DOI: 10.1002/sdr

information may result in shortsighted decisions and unanticipated consequences. Experimental research in system dynamics shows that people typically struggle to understand the relationship between stocks and flows even in the simplest possible stock-and-flow system (Cronin *et al.*, 2009; Sterman, 2010; Sweeney and Sterman, 2000), a reasoning error caused by poor understanding of accumulation is known as *stock–flow failure*.

Mistaking multiplicity of fractions for proportionality of flows

An increase in the inflow rate causes a transient imbalance with respect to the outflow rate, generating disequilibrium dynamics that lead to higher amounts of stock.[ii] Under proper conditions, management faced with such changes would eventually act on the system to stabilize it, raising outflow rates and bringing the system to equilibrium. Under the new equilibrium, Little's Law ($L = \lambda w$) would determine that the equilibrium amount of Stock (L) would be proportional to the value of the new flow through (λ). That is, changes (e.g. an increase or decrease) in inflow rates proportionally affect the Stock (L), or the wait (w), or both. In practice, if Etel were to observe a 10% increase in contracting rates (λ), managers might observe a 10% increase in the order backlog (L) stock (or a 10% increase in the time to activate contracts) (see Figure 5). Without changes to the causes of fallouts, the higher backlog would lead to a 10% increase in the fallout rate ($L\gamma$). Similarly, a 10% change (e.g. increase or decrease) in the fallout fraction[iii] (γ) would proportionally affect the fallout rate ($L\gamma$). Hence, changes in the contracting rate (i.e. inflow rate) or the fallout fraction (i.e. quality) proportionally affect the fallout rate (i.e. outflow). This proportionality can be deceiving to managers, especially when managers compare flow rates (measured in absolutes) with quality levels (measured in percentages). Because managers at Etel allocate human resources (HR) capacity in expectation of fallout rates, failure to assess the impact of changes in the fallout fraction and contracting rate can lead to inadequate HR capacity decisions.

At Etel, managers faced both types of changes in contracting flows and expected contracting process quality. At a time when contracting flows increased by 10% and actual quality dropped 10% (from 90% to 80%), managers expected a total increase in the fallout rate of 20%. So, they adjusted the required HR capacity, increasing it by 20%. They wrongly assumed that a 20% increase in hiring would suffice to address the increase in fallout rate. Focused on the proportionality of flows and quality, managers struggled to

[ii]Understanding that the stock must rise if the inflow exceeds the outflow may not be immediately obvious to managers (Sterman, 2010).
[iii]The fallout fraction can be understood as the inverse level of quality (if the fallout fraction were zero, the associated quality would be 100% – or perfect service).

DOI: 10.1002/sdr

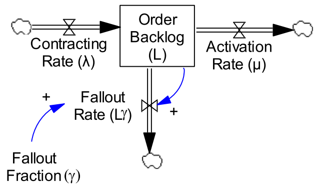
System Dynamics formulation	Queueing formulation
	*Contracting rate* (λ) = *Fallout rate* ($L\gamma$) + *Activation rate* (μ) *Average wait* (w) = *Order Backlog* (L) / *Contracting rate* (λ) $w = \frac{[Contracting\ rate\ (\lambda) - Activation\ rate\ (\mu)]}{[Fallout\ fraction\ (\gamma) Contracting\ rate\ (\lambda)]}$ [*Or*, $w = (\lambda - \mu)/(\gamma\,\lambda)$]

Fig. 4. System dynamics and Queueing formulations with fallouts [Color figure can be viewed at wileyonlinelibrary.com]

understand that they would need to more than double the required capacity to address the influx of problems.

Here's why. If the fallout fraction increases by 10% from the original 10% level (i.e. 90% quality), the new fallout fraction would simply become 11% (i.e. 89% quality). But a quality drop from 90% to 80% actually represents a *doubling* in the fallout fraction (from 10% to 20%). Hence, Etel should have prepared for a doubling of the fallout rate. Such fallout fraction would require twice as much HR capacity as originally planned. Naturally, as the fallout fraction (γ) doubles (from 10% to 20%), so should the required HR capacity. Figure 4 displays the system dynamics and queueing formulations with fallouts.

> *Principle*:
> Management of process outflows may be deceiving due to the proportionality associated with changes in inflows (due to Little's Law) and changes in fractional quality affecting those outflows. This proportionality can be deceiving to managers, especially when managers compare flow rates (measured in absolutes) with quality levels (measured in percentages).

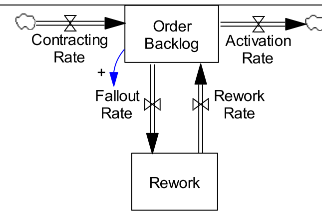
Mistaking the multiplicity of fractions for the proportionality of flows can have challenging implications for managerial decisions associated with the levels of outflows. Previous research (Cronin *et al.*, 2009; Sterman, 2010; Sweeney and Sterman, 2000) shows that people resort to correlational reasoning to explain the relationship between stocks and flows, typically adopting *correlation heuristic*, to conclude "that a system's output is positively correlated with its inputs p. 316" (Sterman, 2010).

Critical stocks and average residence times

All else equal (e.g. same outflows), an increase in fallout rates results in a higher value for the amount of the stock of rework. Because it takes time (e.g. hiring, training) to make additional HR capacity available to handle the

DOI: 10.1002/sdr

System Dynamics formulation



Queueing formulation

$$Ave\ Backlog\ Process\ time\ = \frac{Order\ Backlog}{(Activation\ rate\ +\ Fallout\ rate)}$$

$$Ave\ Rework\ time\ = \frac{Rework}{Rework\ rate}$$

Fig. 5. System dynamics and Queueing formulations for average residence times in critical stocks [Color figure can be viewed at wileyonlinelibrary.com]

increase in rework levels, in the short term Etel would experience an increase in the average time to address rework problems. Etel could keep track of the average rework time by computing the ratio between the stock of rework (R) and the rework resolution rate (Figure 5).

Analogously, an increase in contracting rates would result in a higher value for the stock (e.g. order backlog). If Etel would like to maintain a constant delay to process orders in its backlog, it would need to increase capacity to address those additional contracts. Because it takes time (e.g. hiring, training) to make additional HR capacity available, however, in the short-term Etel would experience an increase in the average time to process the backlog of orders. Etel could keep track of the average backlog processing time, by computing the ratio between the order backlog (L) and the activation rate (μ).

Principle:
Information on critical stocks help managers assess the state of the system and help them evaluate the need for intervention. By keeping track of major stocks and their inflows, and outflows, managers can use Little's Law to assess average residence times for units in those stocks. Residence times provide a useful measure of performance in such systems.

Downstream work feedback on quality

The stock of rework increases the required level of back-office workload, which in turn affects fallouts. That is, a higher level of back-office workload leads to more mistakes and a higher fallout fraction, which creates even higher rework levels, and an even higher workload. In queuing this feedback characterizes queues with state-dependent arrivals, known as a system with discouraged arrivals (Kleinrock, 1975). The process described above closes the reinforcing *Fallout Loop* (R1) captured by the Yerkes-Dobson Law (Sterman, 2000). The reinforcing *Fallout Loop* (*R1*) captures the rework impact

DOI: 10.1002/sdr

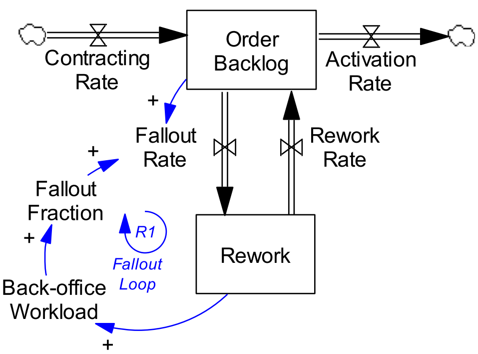


Fig. 6. System dynamics formulation capturing the feedback from rework to fallout fraction [Color figure can be viewed at wileyonlinelibrary.com]

on the fallout fraction which makes the quality problem even more difficult to manage. This feedback mechanism can operate in a *vicious* way when high levels of back-office workload causes a high fallout fraction, leading to even higher rework and workload levels. Alternatively, the loop can operate in a *virtuous* way, when low levels of back-office workload lead to a low fallout fraction, and even lower future workload and fallout levels. Etel has struggled with *R1* operating in a vicious way, requiring significant investments to properly address it. Figure 6 displays the system dynamics formulation capturing the reinforcing feedback from rework to fallout fraction.

Principle:
In a queuing system with discouraged arrivals (Kleinrock, 1975), the amount of items in the queue affects the number of new arrivals. This state-dependent arrivals capture the feedback in the system. This reinforcing feedback mechanism can operate in a *vicious* or *virtuous* way, depending on the level of stock and its impact on arrivals. Typically, as the amount of rework increases, it leads to higher fallout fractions, and even more rework, in a vicious spiral that makes the quality problem very challenging to manage. This insight is not new and has been previously captured in Oliva and Sterman (2001). However, it still represents a sizable challenge for managers. Because it is a reinforcing mechanism, if managers fail to maintain a low fallout fraction (e.g. high quality), the rework problem may get out of control.

Quality cascades and multiplicative effects on downstream processes

The problem with persistent rework is that it affects downstream processes such as call and complaint handling. Etel's managers failure to realize the buildup of rework prevented them from anticipating the impact it would have on the volume of calls and complaints. The stock of rework directly

DOI: 10.1002/sdr

influences the call volume because customers call to solve their problems. But the call volume also depends on the average residence time of orders on rework because customers whose orders have not been reworked call more frequently to solve them. The resulting call workload depends both on the rework volume and also on the rework wait. Hence, rework problems (in the primary process of activation) cascade to generate higher call volumes, increasing the call workload (i.e. problems in the secondary process of call handling). A further cascade generates customer complaints.

Multiplicative effects also occur due to both the number of orders in rework and their average residence time (i.e. the persistence of rework). Consider first the former effect. Customers whose orders must be reworked call the company to try to resolve them, increasing the call volume and the workload on the call center. As the call center communicates those problems to the back office, they not only put pressure to rework them (e.g. resources allocated to rework rate), but also on back-office operations, increasing their workload. The effect closes the reinforcing *Rework Volume Loop* (*R2*) capturing the rework impact on call volume and its influence in back-office workload. Consider next the latter effect. Orders whose problems are not resolved stay longer in the stock of rework. Customers whose orders stay in rework for prolonged periods of time call the company multiple times to try to resolve them. As the rework time increases, so does the frequency of calls and, with it, call volume and call-center workload. As the call center communicates the problems to the back office, they escalate back-office operations workload, closing the *Rework Wait Loop* (*R3*) reinforcing loop. The reinforcing loops capture the multiplicative effects on downstream processes.

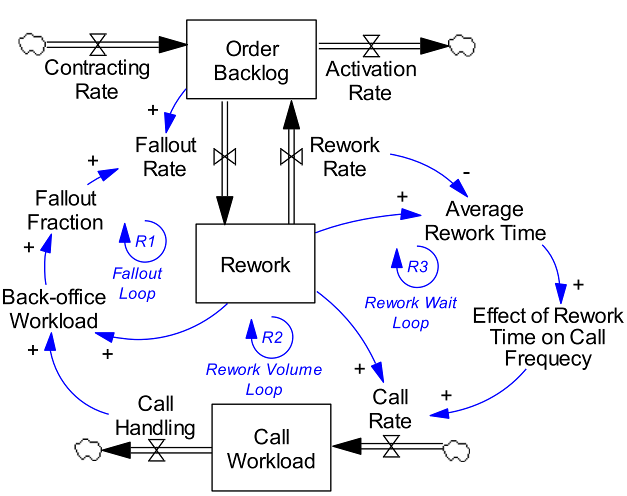


Fig. 7. System dynamics formulation with cascading and multiplicative effects [Color figure can be viewed at wileyonlinelibrary.com]

DOI: 10.1002/sdr

Because decision-making at Etel was heavily compartmentalized, the impact of these quality cascades were exacerbated. In addition, because departments communicated and shared decisions with each other only on a monthly basis, delays compounded the problem. In Etel's case, the cascades were internal to the company, but they can also spill outside affecting partner companies. Figure 7 shows the system dynamics formulation with cascade and multiplicative effects on call rate.

Principle:
Due to functional boundaries, it is typically difficult to identify and plan for cascade effects. High-level mapping of the interconnections between processes can help organizations identify possible areas where problems can cascade and impact downstream processes, or even other companies. It is also important to consider the nonlinear and multiplicative effect that some parameters (e.g. average rework time) can have on others (e.g. HR capacity addressing calls and customer complaints). A nonlinear multiplicative effect cascading across companies is captured in Gonçalves (2018), where long supplier-delivery delays cause retailers to aggressively increase their orders beyond customer demand.

Conclusion

This note distills six principles that combine the strengths of system dynamics and queueing theory and build on the understanding of stocks and flows to help managers (and modelers) generate important insights in practice. Previous research suggests that people often "employ heuristics that are intuitively appealing but erroneous p. 117" (Cronin *et al.*, 2009), such as the *correlation heuristic* and *stock-flow failure*. Training in system dynamics can help. The first two principles, albeit simple and devoid of feedback, provide valuable insight. By recognizing the transient nature of flows (e.g. *inflows and outflows are rarely in steady state)*, managers can identify organizational linkages and inform specific decisions. The second principle highlights the informational value of stocks. By recognizing the cumulative nature of stocks (e.g. *stocks integrate the imbalance between inflows and outflows)*, managers can keep track of an important source of information for managerial decisions.

Previous research also suggests that capturing feedback in queueing models can significantly improve the insights of the resulting models. The four remaining principles, which include feedback, provide more critical, interesting, and counterintuitive insights. By recognizing that proportionality can be deceiving (e.g. *mistaking multiplicity of fractions for proportionality of flows*), managers can make appropriate decisions when dealing with

DOI: 10.1002/sdr

changes in flow rates and changes in quality levels. The next principle (e.g. *critical stocks and average residence times*) combines the value of critical stocks with Little's Law to assess average residence times (i.e. an important measure of performance) for the items in those stocks. By recognizing that some stocks can feed back to influence their inflows (e.g. *downstream work feedback on quality*), managers can realize the presence of reinforcing feedback processes and try to manage the system so it can operate in a *virtuous* way. Finally, by mapping interconnections across different processes (e.g. *quality cascade and multiplicative effects on downstream processes*), managers can identify possible areas where problems can cascade, impacting downstream processes. By recognizing that some effects (e.g. average rework time) have multiplicative impact on others (e.g. HR capacity), managers can better prepare and act preventively to avoid difficult situations.

The six industry examples broaden the cases of simple, insightful, easy to communicate, and impactful models, bridging system dynamics and queueing. The principles seek to generalize the insights illustrated in each example, so they can be useful and applicable to managers operating in different settings. Overall, this work resonates with Homer (1997) that a "potent stock and flow structure" can lead to compelling insights and provide practical examples of flawed decision making associated with correlational reasoning and stock-flow failure. I hope these principles motivate further discussion on ways to improve modeling insights by combining system dynamics and queueing.

Acknowledgement

I am grateful to Henk Akkermans for the invitation to support his work at Etel. I thank Andreas Größler and Markus Schwaninger for their constructive comments and useful suggestions to improve the manuscript. All errors are mine.

Biography

Paulo Gonçalves is Professor of Management and Director of the Humanitarian Operations Group at the Università della Svizzera italiana (USI), Switzerland. He is also Research Fellow at the University of Cambridge Judge Business School (CJBS). His research combines system dynamics simulation, behavioral experiments, optimization, and econometrics, to understand how managers make strategic, tactical and operational decisions in humanitarian settings. Paulo holds a Ph.D. in Management Science and System Dynamics from the MIT Sloan School of Management and a M.Sc. from the Massachusetts Institute of Technology (MIT).

DOI: 10.1002/sdr

References

Akkermans H. 2018. *Service Operations Dynamics: Managing in an Age of Digitization, Disruption and Discontent*. Switzerland: Springer.

Akkermans, H., C. Voss, R. van Oers, Q. Zhu. 2016. Never the twain shall meet? Simulating Sales & Operations Planning ramp-up dynamics in IT-enabled service supply chains. Proceedings of the International System Dynamics Conference.

Barlas Y, Özgün O. 2018. Queuing systems in a feedback environment: continuous versus discrete-event simulation. *Journal of Simulation* **12**(2): 144–161.

Cronin MA, Gonzalez C, Sterman JD. 2009. Why don't well-educated adults understand accumulation? A challenge to researchers, educators, and citizens. *Organizational behavior and Human decision Processes* **108**(1): 116–130. https://doi.org/10.1016/j.obhdp.2008.03.003.

Ghaffarzadegan N, Larson R. 2018. SD meets OR: a new synergy to address policy problems. *System Dynamics Review* **34**: 327–353. https://doi.org/10.1002/sdr.1598.

Gonçalves P. 2018. From boom to bust: an operational perspective of demand bubbles. *System Dynamics Review* **34**: 389–425. https://doi.org/10.1002/sdr.1604.

Homer J. 1997. Structure, data, and compelling conclusions: notes from the field. *System Dynamics Review* **13**(4): 293–309.

Kleinrock L. 1975. *Queuing Systems, Vol. 1: Theory*. Wiley: New York.

Lane DC. 1994. With a little help from our friends: how system dynamics and soft OR can learn from each other. *System Dynamics Review.* **10**: 101–134.

Oliva R, Sterman JD. 2001. Cutting corners and working overtime: quality erosion in the service industry. *Management Science* **47**(7): 894–914.

Shortle JF, Thompson JM, Gross D, Harris CM. 2018. *Fundamentals of Queueing Theory*. New Jersey: John Wiley & Sons.

Sterman JD. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Irwin/McGraw-Hill: Boston, MA.

Sterman JD. 2010. Does formal system dynamics training improve people's understanding of accumulation? *System Dynamics Review* **26**: 316–334. https://doi.org/10.1002/sdr.447.

Sweeney LB, Sterman JD. 2000. Bathtub dynamics: initial results of a systems thinking inventory. *System Dynamics Review* **16**: 249–286. https://doi.org/10.1002/sdr.198.

DOI: 10.1002/sdr