
Article

Three generic policies for sustained market growth based on two interdependent organizational resources – a simulation study and implications

Martin Schaffernicht ¹

¹ Facultad de Economía y Negocios, Universidad de Talca (Chile); martin@utalca.cl

Abstract: This article addresses the generic dynamic decision problem of how to achieve sustained market growth by increasing two interdependent organizational resources needed (1) to increase and (2) to sustain demand. The speed and costs of increasing each resource are different. Failure to account for this difference has been reported to lead to policies that drive a quick increase of demand followed by decline. Three generic policies derived from the literature have been implemented in a system dynamics model. Simulation shows that all three policies can generate sustained exponential growth but differ in performance. These results suggest that even policies which risk generating overshoot and collapse can avoid it. This poses two questions for further research: (1) what is the reasoning of human decision-makers when choosing between these policies and (2) how can the important but easily overlooked features of such decision situations be made sufficiently salient to be accounted for?

Keywords: dynamic decision-making; interdependency; organizational resources; overshoot and collapse; simulation

1. Introduction

This article proposes a theoretical examination of a dynamical problem faced by decision-makers in charge of developing organizational resources which are relevant for growth. The relationship between resources and growth has been problematized in the literature of the resource-based view of strategy [1-6]. An essential idea in the resource-based view is that an organization's performance depends on the availability of resources - material or immaterial - required for processes. Managers must make sure that sufficient resources are available at any time [7]. For growth-oriented organizations, this implies the need to develop such resources into the future [2, 8-10]: the stock of these resources must increase over time.

The dynamic nature of this task is already challenging due to the interdependencies between them, but its complexity increases when different resources grow at different speeds [11]. In such cases, their trajectories react to one another in counter-intuitive ways: decision-makers build mental models of the situation's structure which contain flaws like the "misperception of feedback" [12-15] or mentally infer incorrect behavioral implications like in the case of the "stock and flow error" [16-22]. Such flawed mental models lead to policies that drive the system into an initial phase of strong and exponential growth followed by a crisis and decline often referred to as "boom and bust" [23]. "Boom and bust" is a special case of "overshoot and collapse" - a type of causal structure that captures problems of sustainability [23-27]. Such unsustainable decisions have not only been observed in real-life situations, but also in very simplified laboratory situations involving only a reduced number of resources and interdependencies [12, 13]. Reducing the number of features to a minimum needed to inquire a specific type of phenomenon has allowed to make the case of "misperception of feedback", and here, it allows to crit-

ically examine the generic decision policies that have been discussed in this area of research.

The simplest possible case of interdependent organizational resources consists of two internal and one external resource. The first internal resource is used to increase *demand* and represents the variety of marketing and advertising activities; hereafter it will be referred to as *salesforce* to avoid unnecessarily abstract language. The second internal resource – *production capacity* – summarizes all that is necessary to deliver according to the *demand*, so that customers will not be deceived and shift their *demand* to competitors. The third resource is external: *demand* is a sign of the customers' *satisfaction* – it cannot be directly controlled by the organization, but it is necessary to keep the organization going [28]. The case also includes *financial resources*, but since they are a consequence of *demand*, they do not receive a specific treatment of their own.

If the *salesforce* can grow faster than the *production capacity*, it may be intuitive to use the *salesforce* as a driver for growth and adjust the *production capacity* as necessary. For many small and medium enterprises – especially in the developing world – the investments necessary to increase *production capacity* are huge and risky: one must mobilize *financial resources* incurring and long-term obligations without having the certainty of a sustained growth in *demand*. However, a policy which increases the *salesforce* and postpones increasing *production capacity* until the increased *demand* puts significant stress on the *production capacity* and carries the risk of failing to adjust the *production capacity* on time, deceiving customers and then facing a decreased *demand* when eventually the *production capacity* has grown [29, 30].

For growth oriented companies which are not afraid of this risk, have huge fundraising possibilities or important capital reserves, early and substantial investment in capacity has been recommended because of several reinforcing effects enabling sustained competitive advantage; at the same time, there is a thin line between getting big fast and getting too big too fast [24, 31-33]. The importance of developing sufficient *capacity* has been stressed [31], but “overshoot and collapse” as well as other cyclic instabilities continue to challenge managers [24, 34]. The delicate balance between the firm's resources is still an important issue management education [30].

The perception and relative importance attributed of (1) the risk of long-term depth and financial hardship and (2) the risk of failing to increase the *capacity* on time will depend on the particular context of an organization's founder or manager. Some may perceive the first risk more clearly than the second one, others may perceive both risks but value them differently. The studies concerned with the “misperception of feedback” suggest that the second risk is not clearly perceived due to the time delays.

When prompted to choose one of these two policies, naïve individuals (without specialized knowledge) may also suggest that they would rather not use *salesforce* or *production capacity* as growth driver and adjust the respective other resource, but instead drive the increase of both resources in the same way maintaining the necessary synchronization. This leads to three different kinds of logic or policies:

Policy P1: Drive growth through increases of the *salesforce* and adjust *production capacity* as needed.

Policy P2: Drive growth by investing in *production capacity* and adjust *salesforce* as needed.

Policy P3: Drive growth through simultaneous and harmonic increases of *salesforce* and *production capacity*.

Each of these policies has been implemented in a simulation model inspired by a classic system dynamics model which is prominent in system dynamics and management education [29, 30]. Comparison of the resources' behavior over time and the performance in terms of growth reveals that all three policies can generate an itinerary of sustained exponential growth, albeit policy P2 allows to reach a stronger acceleration and much higher performance. This suggests that neither of these policies is condemned to

produce a “boom and bust” pattern: it can be avoided if decision-makers are able to implement the chosen policy with carefully chosen parameter values.

The remainder of this article starts with a section which introduces a stylized decision situation in the context of the three resources and the relationships between them, outlining the three generic policies. The ensuing section presents the specific features of the decision situation used in the simulation model and compares the results obtained by each of the policies. Based on these results, the fourth section proposes a revision of how the generic policies have been interpreted in previous studies, leading to a series of questions for empirical research.

2. Theory: a three-loop system

2.1 The structure and dynamics of a three-resource system

Consider a system consisting of two agents: a company and customers. The company supplies a product that customers need or want, and the customers compensate the company with a payment, which is what the company needs or wants. Each of the two agents has something the respective other agent wants and wants something that the other agent has. Both will interact as long as this relationship is reciprocal. If the company has been recently created, the relationship does not exist yet because the customers do not know that the company can supply something they want. Therefore, one of the challenges for the company is letting the customers know and desire the product - and the necessary processes will only take place if the company has sufficient *salesforce* to fuel them. Assuming these processes are successful, the company will perceive an increasing *demand* for their products. Now, this growing *demand* must be satisfied, and again, the necessary processes will only take place if the company has sufficient *production capacity*. Failing to produce and deliver will lead to a decrease of *demand*.

This fundamental structure can be conceptualized as a dynamic system like illustrated in Figure 1, where each of the agents is represented as a dotted rectangle: the company (left) and the customers (right). The figure contains a causal loop diagram [35, 36] consisting of feedback loops, variables, and causal links which are part of the decision situation regardless of decision-makers' policies. Loops are marked by a circular symbol containing a letter (“R” for reinforcing feedback and a “B” for balancing feedback). Two kinds of variables are visually distinguished: stocks appear in solid rectangles, and other variables are included simply with their name. The arrows between variables are causal links with a polarity sign and a delay mark where the relationship between two variables is delayed with respect to the other relationships. Positive polarity means that when the independent variable increases/decreases, the dependent variable's behavior will change upwards/downwards. When an increase/decrease in the independent variable leads to a downward/upward change of the dependent variable's behavior, polarity is negative.

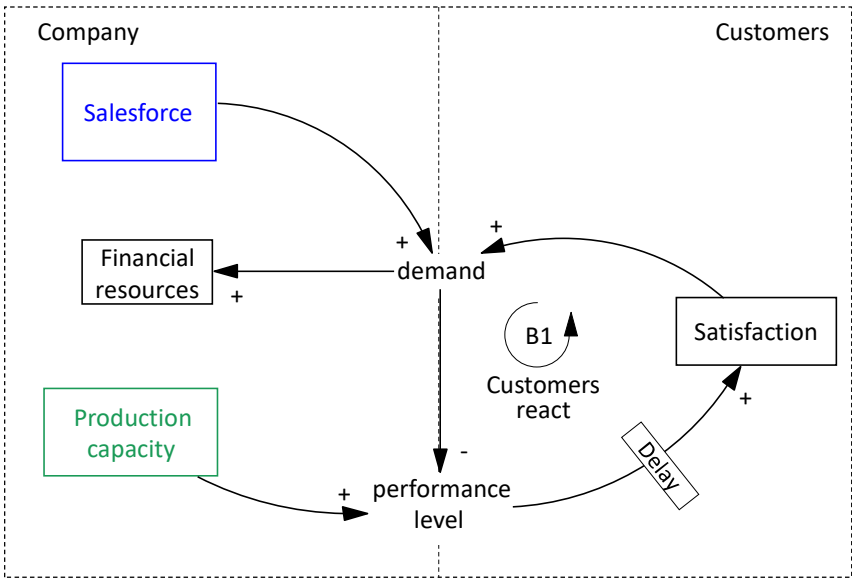


Figure 1: The fundamental structure of the system with two critical internal resources

In Figure 1, both agents interact through two variables: the customers signal their *satisfaction* through their *demand* and perceive the company’s *performance level*. Loop B1 represents the customers’ reaction to the *performance level* they perceive. The process of perceiving consists of gradually adjusting the perceived level to new signals over time. Accordingly, changes in the *performance level* will take some time to trigger the corresponding changes in the level of *satisfaction*, which will slowly respond to an increasing or decreasing *performance level* by an increase or a decrease of *satisfaction*. Increasing *satisfaction* leads to increasing *demand*, just as a decrease in *satisfaction* triggers a decrease of *demand*. The relationship between *demand* and *performance level* is different: an increase in *demand* will lead to a decrease of the *performance level*, given the current *production resources*. This also means that a decrease of *demand* helps to increase the *performance level*. Therefore, when customers turn away from a company and the *demand* decreases, this is one way to correct problems in the *performance level*. However, from the standpoint of the company, this is an external correction – it cannot be directly controlled - a loss is the opposite of what a growth-oriented company aims for.

Inside the company, increasing *demand* leads to an increase in *financial resources* and decreasing the *demand* leaves the company with less *financial resources* than would have been the case otherwise. Other than this relationship and the causal influence of *production capacity* on the *performance level* and of the *salesforce* on *demand*, nothing is specified: the way how the company reacts to changes in its *financial resources* and how it will steer the two resources driving its relationship with its customers will be specified by the policies. The three different possibilities for steering each of the two resources are three different ways to connect *salesforce* and *production capacity* to *demand*, thereby creating two feedback loops connected to the “customers react” loop. Growth drivers are self-reinforcing processes based on reinforcing feedback loops, whereas adjustments are control processes driven by balancing feedback loops [36]. Therefore, the policies are formulated as three different pairs of feedback loops, as shown in Table 1:

Table 1: The three policies are based on three different pairs of feedback loops around *salesforce* and *production capacity*.

Policy	Resource	
	<i>Salesforce</i>	<i>Production capacity</i>

P1	Reinforcing	Balancing
P2	Balancing	Reinforcing
P3	Reinforcing	Reinforcing

Regardless of which policy steers *salesforce* and *production capacity*, both resources are interdependent. An increase of the *salesforce* stronger than the increase of *production capacity* will put stress on *production capacity*, making it ever more urgent to increase *production capacity* because the additional *demand* decreases the *performance level* and risks to lead to decreasing *satisfaction* and *demand* soon. An increase of *production capacity* stronger than the increase of the *salesforce* will have two consequences: (1) it puts stress on the current *salesforce* to generate the additional *demand* which can now be attended to, and (2) it makes it easier to recruit additional salespeople because the increased *capacity* will cause *satisfaction* to increase, which in turn increases *demand* and leads to increased sales revenues, thereby also increasing *financial resources* and the budget for *salesforce* salaries.

2.1. Different policies to develop resources in this context

2.1.2. Policy P1: Drive demand growth with the *salesforce* and adjust *production capacity* when needed

Policy P1 consists of one set of rules for *salesforce* development which operate as a reinforcing feedback loop which we will refer to as R1, and another set of rules for developing *production capacity* which is a balancing loop with the identifier B2.

The reinforcing loop “increase demand” (R1) contains the positive link from *demand* to *financial resources*. Any change the stock of *financial resources* triggers a change with the same sign in the *salesforce*: more money automatically leads to more salespeople and decreasing *financial resources* entail decreasing *salesforce*. Changes in the *salesforce* cause changes of *demand* with the same sign: an increased or decreased number of salespeople always leads to an increase or decrease of *demand*. This loop is reinforcing and will always accelerate change in the direction of increase or decrease unless this behavior is limited by other factors. However, there are two more loops connected to it – directly or indirectly, as illustrated in Figure 2:

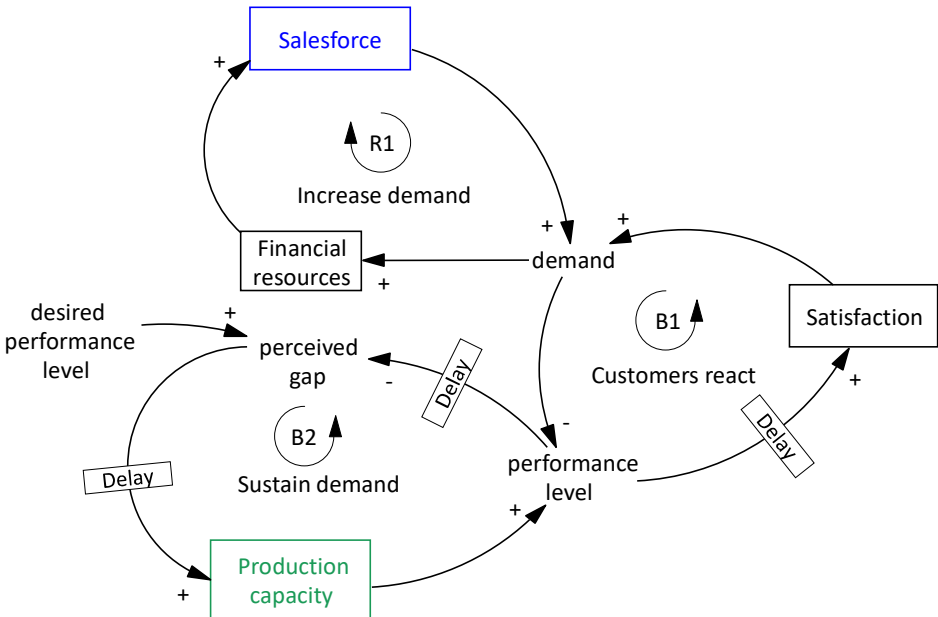


Figure 2: *Salesforce* drives *demand* and *production capacity* is adjusted.

Figure 2 also shows the “sustain demand” loop (B2), which is internal to the company and sensitive to the *performance level*. The company compares the available observations of the *performance level* to a desired *performance level*. The delay between actual *performance level* and its perception by the company is due to regular administrative processes. If the company perceives a gap between the desired and the observed *performance level*, it will adjust its *production capacity*. For instance, if the *performance level* has decreased, then the perceived gap increases, and therefore *production capacity* will be added to current stock. Increases in the observed *performance level* will reduce the gap and it is even possible that a surplus of *production capacity* constitutes a negative gap, leading to a reduction of *production capacity* – a way to reduce costs that are apparently unnecessary. As argued earlier, this can be perceived by the decision-maker as less risky than making huge investments in *production capacity* before *demand* actually increases. In such a case, it seems logical for the company to be cautious with the *financial resources* and therefore to only correct decreases in the *performance level* by increasing the *production capacity*.

Any additional *production capacity* must be ordered, constructed, mounted, and put into service before it can be used: there will be a sizable delay. As soon as the additional *production capacity* goes online, the company’s *performance level* will increase (momentarily assuming an unchanging *demand*). This is a balancing loop which will control the level of *production capacity* such as to keep *performance level* close to the desired *performance level*.

The “sustain demand” loop (B2) is linked with the “customers react” loop (B1) by the *performance level*. Both loops use one variable to control the *performance level*. The company needs to correct *performance level* gaps before customers will adjust their *satisfaction level*: if the *satisfaction level* can decrease quicker than the *production capacity* can be increased, internal corrections to the *production capacity* will come too late. Also, note that there will be no increases of the *production capacity* unless a gap is perceived. This can only happen when *demand* grows faster than the *production capacity* – for a constant level of these resources, any increase of *demand* has this effect. It can even happen that a strong reaction of the customers, entailing a strong decrease of *demand*, increases the *performance level* so strongly that the company erroneously detects a need to decrease its *production capacity*. Since such a decrease of *production capacity* might also decrease the *performance level*, a reinforcing process might take over: combined, loops B1 and B2 are a reinforcing feedback loop.

Overall, the system which is constituted by this policy risks to severely underperform in terms of growth and even develop into a decline or collapse, as has been shown in previous publications with specific examples of this generic case [29, 30].

2.2.2. Policy P2: Enable *demand* growth through *production capacity* and adjust the *salesforce* when possible

One way to avoid the vicious side of the first policy is to prioritize the increases of the *production capacity* and thereby assure that the effect of an increasing *salesforce* will not unintentionally lead to a decrease of the *performance level*. This is a different approach to sustaining *demand*, and it is shown in the following Figure 3.

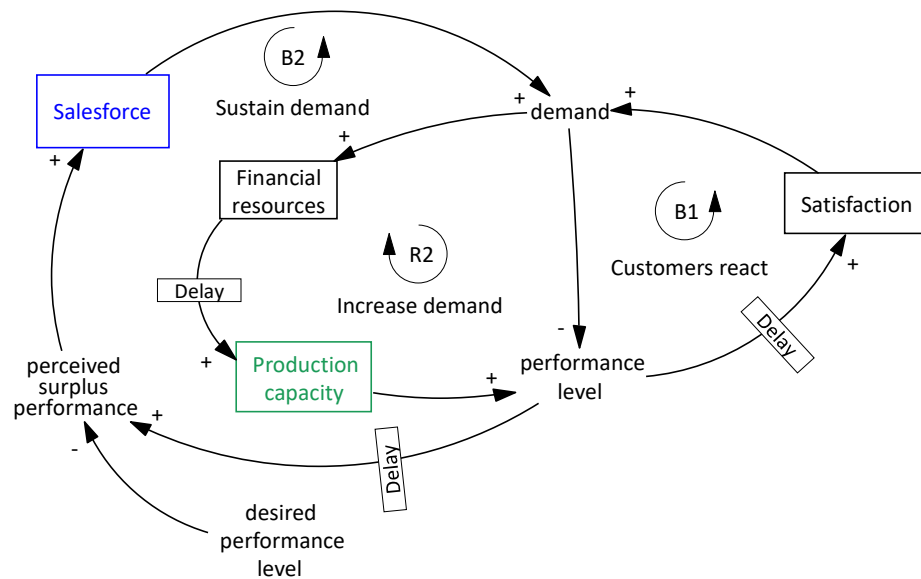


Figure 3: *Production capacity drives demand and salesforce is adjusted.*

The causal loop diagram in Figure 3 contains the same basic structure (loop B1 and *salesforce* impacting the *performance level* through a positive causal link), but the logic of steering the internal resources is inverted. Changes to the *financial resources* drive the development of *production capacity*, whereas the *salesforce* reacts to *perceived surplus performance* - a gap between the observed and the desired *performance level* - such as control the *performance level*. In this configuration, the “sustain demand” loop adjusts *salesforce* and the “increase demand” loop drives *production capacity*.

This configuration will increase the *salesforce* only when an increase of the *production capacity* has already increased the *performance level*, which means there is a surplus *performance capacity* which will remain unnecessary until additional *salesforce* can increase the *demand* such as to press *performance level* back to its usual value. Where policy P1 will be late to adjust the *production capacity*, policy P2 accepts the effect of spending “too much” money for *production capacity* - as compared to the level of *demand* - a sacrifice for being able to produce sufficiently once the increased *salesforce* has led to a *demand* which with fully require the increased *production capacity*. The fact that the “customers react” loop is quicker to impact *performance level* than the “sustain demand” loop is not a danger here because changes to *satisfaction* with only be triggered when change the *production capacity* have already happened. For decision-makers who do not mind the perceived risk of being unable to increase *demand* is sufficiently, being able to avoid a loss of *satisfaction* caused by insufficient *production capacity* will make policy P2 attractive.

So, even though P2 can generate sustained and accelerating increases of *demand*, it also contains the possibility of an accelerating decrease.

2.2.3 Policy P3: drive and enable growth simultaneously

Decision-makers who are aware of the risk inherent in policy P1 and find themselves unable or unwilling to make important investments ahead of possible increases of *demand* (P2) may consider a third possibility: use increases in *financial resources* to simultaneously increase the *salesforce* and *production capacity*. This logic is shown by the causal loop diagram in the following Figure 4:

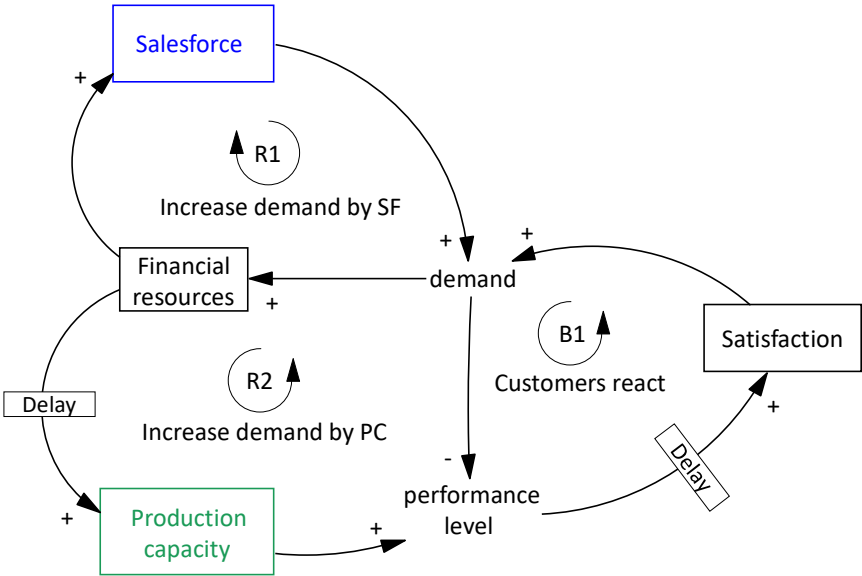


Figure 4: *Salesforce and production capacity simultaneously drive demand.*

The structure shown in Figure 4 conserves the “customers react loop”. The causal link from *demand* to *performance level* belongs to two different loops, and so does the link from *demand* to financial resources. Now, both loops internal to the company are reinforcing in nature. The company’s logic of steering the *salesforce* is the same as in policy *P1*, and its way to steer *production capacity* is the same as in policy *P2*. Since “increase demand by SF” (R1) as well “increase demand by CP” (R2) are reinforcing, it can indeed happen that the potential decrease of *performance level* in response to an increased *demand* is exactly compensated for by the effect of additional *production capacity* on the *performance level*. Therefore, *demand* could indeed grow quickly – accelerating - and be sustainable. Achieving such steady and accelerating growth requires that the company be able to determine which fraction of additional financial resources is used for *salesforce* and which fraction is to be used for *production capacity*. Of course, such a growth would still be limited by overall size of the market and the market share captured by the company. However, this limitation is the same for all three policies, and unless the limit is reached, is theoretical existence does not make a difference through the choice between these three policies.

3. Method

3.1. Description of the specific decision situation used in the simulation

To assess the behavior of the resources under the three policies and their respective performance using a simulation model, some specific features are defined now. This subsection introduces the general aspects; the policies discussed here can be interactively examined as described in Appendix A, and the exact formulation of the simulation model is reported in Appendix B.

Decision-makers take the role of a manager who has been recruited by a young company which produces and distributes customized watches. The manager will be responsible for a new division which has been launched recently with a planning horizon of around 10 years. The short-term goal is to achieve a strong and sustained growth in terms of sales over a period of 50 weeks. When taking office, there are 40 salespeople and up to 1,512 weekly watches can be produced.

For the sake of simplification, customers are assumed to care only for the *delivery delay* - the number of weeks between placing an order and the delivery of the watch -

with a *delivery delay* of two weeks. This *delay* assures that salespeople can sell at 90% of their maximum productivity: 9 new orders per week. The company might increase customers' willingness to buy to 100% by becoming quicker to deliver, but the loss of *demand* caused by becoming slower to deliver is more drastic. It follows that the price of € 431 does not bother customers, and therefore the price is kept constant during the entire game. The initial weekly *production capacity* of 1,512 watches (with 21 shifts) assures that the *open orders* – 720 at the beginning – can be delivered in this delay, and the *salesforce* of 40 individuals will be able to produce 360 *new orders* per week.

Each week, several decisions must be taken. The first is the recruitment or layoff of salespeople. Recruiting salespeople takes somewhere between one and four weeks depending on the number of salespeople to be recruited. To simplify, € 3,000 is the standard salary for all salespeople, and they all have the same *productivity*, being able to bring in a maximum of 10 *new orders* per week. The *salesforce* is limited by a budget rule: not more than 25% of *expected sales revenue* can be used for *salesforce* salary. If there has been an increased number of *deliveries*, 25% of the additional *revenue* will be available for recruitment, and the manager decides which percentage of this to use.

For production (including delivery), the weekly number of *producible* watches depends on the *production capacity* and on the *capacity utilization fraction* (CUF), which is the actual weekly number of shifts divided by the maximum weekly number of 21 shifts (3 shifts for 7 days): the number of daily shifts times the number of labor days per week. Late or night shifts and work on weekends cause increased salary costs. But modulating the CUF allows to increase effective production without investing in additional *production capacity* up to a certain point. There is a *capacity construction delay* of eight weeks: capacity orders in week w will only come online in week $w+8$.

There's also a budget limit for spending money on capacity increases: participants are informed of a maximum weekly amount and can decide which percentage of this to use. This additional capacity spending is determined by the difference between the *production capacity* needed to produce and deliver 50% of the *open orders* (such as to assure delivery in two weeks) and the current *production capacity*, multiplied by the unit cost of *production capacity*. Note that this allows the manager to order all additional *production capacity* needed to maintain the *delivery delay* – or less. Additional *production capacity* for 1 weekly watch costs € 20,000 and works for 8 years, which is equivalent to 416 watches. This makes roughly € 48 per watch. The *total costs* for producing and selling one watch range from € 283,33 to € 333,33, depending on the *capacity utilization fraction*.

At the sales price of € 431, there remains a difference of between € 147,67 and € 97,67 per watch, so over the lifespan of the production equipment, *profits* are assured. However, it is impossible to fully compensate the initial *investment costs* by *sales revenues* in as little as one year.

3.2. Benchmarks for comparing possible policies

3.2.1 The classical example of market growth and underinvestment in capacity

The case used here is a simplified version of the “market growth and underinvestment” case [29, 30], which dealt with a company producing navigation devices for boats and had a time horizon of 96 months (8 years). This company followed policy *P1*. In comparison to our case, it included a fourth feedback loop representing eroding goals because the workers in the production facility get used to longer *delivery delays* and do not feel the urgency to produce in less time. Additionally, when management demands funds for additional *production capacity*, headquarters discounts a certain fraction. The underlying simulation model generates an initial episode of growth which quickly turns into a steady decline, as shown in Figure 5 (graphic elaborated by the authors based on the model published in [30]):

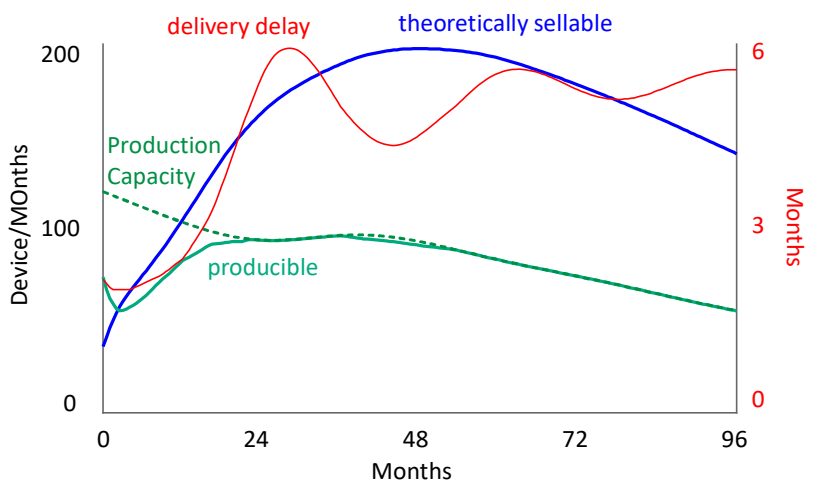


Figure 5: *Salesforce*, *production capacity* and *delivery delay* in the classical example.

Figure 5 shows the monthly number of devices the salespeople can sell assuming their regular productivity and the monthly number of devices the company can produce. The *production capacity* is a dotted line, and together with the *CUF* (not included in the graph) it determines what is actually *producible*. The key features of this behavior are the rise of the *delivery delay* and its oscillations, the decrease of the producible quantity of devices and the boom and bust of the sellable quantity of devices.

When the fourth loop (eroding goals) and the discount of funds for additional *production capacity* are taken away the behavior changes as displayed in Figure 6 (values on the vertical axis print “1k” for 1,000):

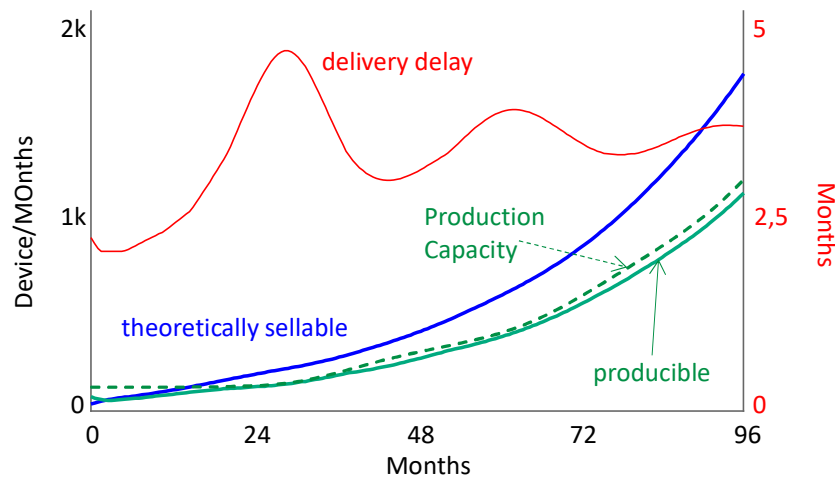


Figure 6: Behavior of the “market growth” model without the “eroding goals” loop and with full capacity investment allowance.

As shown in Figure 6, the *delivery delay* still oscillates and it still increases to somewhere near 3,5 months, which is better than the initial 6 months but still is not what the company wishes to achieve. The decrease of the producible quantity has not turned into a moderate exponential growth. The sellable quantity also grows exponentially - with minor oscillations along its path - and at the end of the 96 months, the *salesforce* could reach up to 1,700 monthly *new orders*, while around 1,000 monthly watches are *producible*.

Even in these conditions, the policy fails to steer the company on an itinerary where *delivery delay* = *desired delivery delay*. The increased *delivery delay* turns customers off and a constant number of salespeople will generate less *new orders*, implying increased *salary costs* per unit sold.

3.2.2. How to include the interdependency between salesforce and production capacity in policies

The policy embedded in the “market growth” model is the substrate of observations made by Forrester [29] in actual companies. Even if one can assume the decision-makers in these companies were aware of the fact that it takes more time to adjust the *production capacity* that the *salesforce*, it is unknown how exactly they took this into account in their decisions. In contrast, the three policies discussed here exploit facts known from the case description in a transparent way.

The stress put on the *production capacity* due to an increasing *salesforce* can be used to determine the recommendable quantity of additional *production capacity*, based on the idea that *production capacity* should be sufficient to produce 50% of the open orders per week to assure that the *desired delivery delay* will not be more than 2 weeks. For this, assume that *total production capacity* = *production capacity* + *capacity under construction*: taking additional capacity that has been ordered but is not yet online is necessary to avoid oscillations [14].

The reasoning is represented by the five following simple equations:

$$\text{desirable deliveries} = \text{open orders} / \text{desired delivery delay} \quad (1)$$

$$\begin{aligned} \text{theoretically producible} = \\ \text{total production capacity} * \text{capacity utilization fraction} \end{aligned} \quad (2)$$

$$\begin{aligned} \text{capacity fractional effort} = \\ (\text{desirable deliveries} / \text{theoretically producible})/2 \end{aligned} \quad (3)$$

$$\begin{aligned} \text{capacity indicated by effort} = \\ \text{total production capacity} * (1 + \text{capacity fractional effort}) \end{aligned} \quad (4)$$

$$\begin{aligned} \text{capacity adjustment indicated by effort} = \\ \text{capacity indicated by effort} - \text{total production capacity} \end{aligned} \quad (5)$$

A stress level of, for instance, 5% would suggest that 5% additional *production capacity* needs to be ordered. Decision-makers can take such a recommendation into account, but they are not forced to fully do so: it can be applied partially as well as fully. This reasoning can be used in different ways, as will be discussed in the following subsections.

3.2.3 Application of policy P1

A player who chooses policy P1 would want to use 100% of the available budget for reinforcing the *salesforce* and order additional *production capacity* as needed. “As needed” needs to be specified in detail. One intuitive possibility is to react to resulting increases in the *delivery delay*, but this is a reactive tactic which requires the decision-maker to wait until the *delivery delay* increases before it can be corrected. The abovementioned possibility calculates the additional *production capacity* required to produce the additional new orders to be expected due to the new salespeople.

Note that both alternatives make no mention of the different delays for actually becoming able to use the additional *salesforce* and the additional *production capacity*. The question arises if even the second possibility – presumably less reactive – implies the risk of overselling and underproducing. For instance, if an additional 20% salespeople are sought for – an increase from 40 to 50 individuals – this will only take one week. As soon as these new salespeople place their 90 weekly *new orders*, 45 additional units of *production capacity* are needed. But if they have been ordered together with the additional *salesforce*, seven more weeks will pass before these capacity units come online. In the meantime, 630 additional *new orders* would have been added to the *open orders*, progressively increasing the *delivery delay*. Since the perception process of customers takes less time than building the new *production capacity*, *demand* would already have diminished, so that there would be less than 630 additional *new orders*: the “customers react” loop would correct the problem of insufficient *production capacity* before the company is able to do so.

Additional complexity comes from the fact that increases of the *capacity utilization fraction* also can absorb a certain amount of additional *new orders*. But as the weekly number of shifts approaches 21, this possibility melts away – so it can be used as a short-term absorber of the relatively sluggish growth speed of *production capacity*, but not as a replacement for investments in additional capacity.

However, simulation of policy *P1* reveals that this risk is avoided. If the decision-maker (1) always uses the full *salesforce* recruitment budget and (2) always orders additional *production capacity* such that the resulting *total production capacity* would be sufficient to produce and deliver all *open orders* within the regular *delivery delay* (assuming 100% *capacity utilization fraction*), then sales, *salesforce* and *production capacity* would display a slightly accelerating (exponential) growth at a constant *delivery delay* of two weeks, as shown in Figure 7.

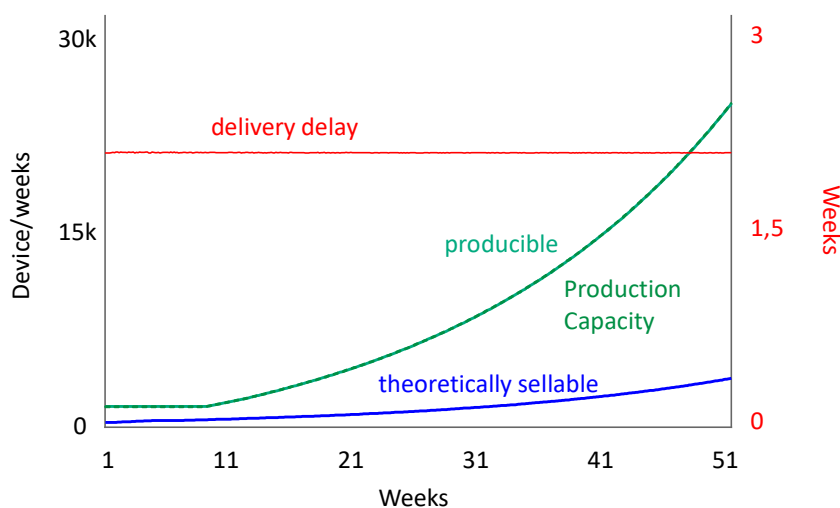


Figure 7: The behavior of the resources under policy *P1*.

Figure 7 also shows that the *production capacity* exceeds the quantity of *theoretically sellable* watches, and increasingly so. The specific behavior and numeric values displayed depend on a *policy implementation fraction*: how strongly the decision-maker converts the recommended change of *production capacity* in actual orders of additional capacity. On a scale from 0 to 1, a value of 0.2 leads to the lowest amount of overcapacity without diminishing the total number of watches sold (0.1 leads to underinvesting in *production*

capacity and would not be sustainable after the end of the year; the simulator described in Appendix A allows to interactively explore the behavior and performance according to varying values of the *policy implementation fraction*). The company would therefore not need to run all 21 weekly shifts ($CUF = 100\%$) to produce sufficiently.

At the end of the year, 398 individuals would be in the *salesforce*, being able to sell 3,585 watches, and the *production capacity* of 5,930 would be 2,344 watches or 65% higher than necessary (assuming $CUF=100\%$). The company would have delivered 66,283 watches; as compared to the 17,280 watches sold without sales growth, this is more than 1.8 times more and corresponds to a monthly growth rate of 9%. The accumulated overcapacity represents all capacity units that were in excess of what the salesforce can sell over the 50 weeks, amounts to 68,859 watches. Overcapacity implies that the company has invested more *financial resources* in *production capacity* that what would have been necessary to fulfill the *open orders* in the *desired delivery delay*. Together with the accumulated deliveries, it allows to judge the policy's performance in comparison to alternative policies.

The stream of recruitment decisions driving this development is a continuous curve for *production capacity* and zigzagging for *salesforce*; this is a consequence of binding *salesforce* recruitment to a constant fraction of *sales revenues* that can rise one week and then fail to keep rising the following week because *production capacity* could not be increased immediately. This is displayed in Figure 8:

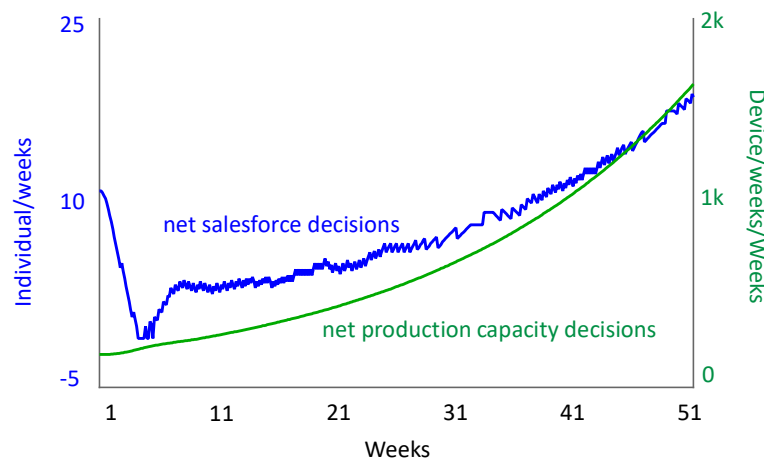


Figure 8: Adjusting the *salesforce* and the *production capacity* under policy *P1*.

3.2.4. Application of policy *P2*

Players who chose policy *P2* will want to apply 100% of the possible increase of *production capacity*, and when the additional capacity comes online – or when it is already online, and the *delivery delay* sinks below one week – recruit additional salespeople. In that case, recruitment is driven by how much bigger the *salesforce* could be without becoming unable to fulfill the *open orders* in the *delivery delay* of 2 weeks. The same idea of putting stress on *production capacity* discussed above is not applied to the salesforce adjustments:

$$\text{theoretically producible} = \text{total production capacity} * \text{capacity utilization fraction} \quad (6)$$

$$\begin{aligned} & \text{theoretic excess production capacity} = \\ & \text{theoretically producible} - \text{theoretically sellable} \end{aligned} \tag{7}$$

$$\begin{aligned} & \text{additional salesforce for excess productibility} = \\ & \text{theoretic excess production capacity} / \text{sales productivity} \end{aligned} \tag{8}$$

This will avoid over-recruiting in comparison to policy *P1*. However, the upper limit for additional *production capacity* depends on the relationship between the current *total production capacity* and the one needed according to the number of *open orders*. So, if there is no or only little growth in *new orders*, only little *additional production capacity* can be ordered. Therefore, it will be necessary to use a certain percentage of the allowed additional recruitments to assure just enough growth of *new orders* to obtain the possibility to make more investments in the growth of *production capacity*. Mentally computing a convenient percentage may be easier when one assumes 21 weekly shifts (*CUF* = 100%), but even so this task is cognitively taxing.

As in the case of policy *P1*, there is a *policy implementation fraction*: if set = 0.7, the development of overcapacity is minimized, and the results are similar shaped than the ones of policy *P1*, but at a much higher absolute level. In Figure 9, the strong growth requires adjusting the vertical axis' scale from 30,000 to 200,000:

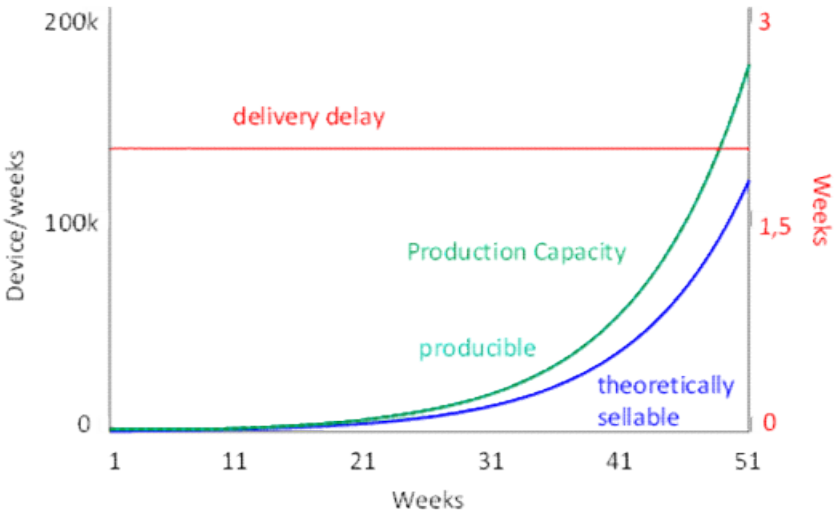


Figure 9: The behavior of the resources under policy *P2*.

Figure 9 shows that the *delivery delay* is held stable at 2 weeks. The *salesforce* reaches 12,656 individuals, being able to sell 113,907 watches for a *production capacity* of 118,860 watches per week. This means only 4% overcapacity at the end of the year (implying the company must stick to 21 weekly shifts), and the *accumulated overcapacity* is 65,959. The *accumulated deliveries* amount to 823,596 watches, which is almost 50 times the baseline sales and implies a monthly growth rate of 38%. Even for a company that is small for the overall market size, it can be doubted if such a growth would not attract competitors, adding a limiting factor to the system; remember, however, that the assumptions given to the decision-makers contain the statement that there is no competitor and overall market size does not limit the company's growth. The stream of decisions concerning *production capacity* is as smooth as for policy *P1*, but policy *P2* also generates a smooth stream of *salesforce* recruitment decisions.

3.2.5. Application of policy *P3*

Decision-makers may use policy *P3* in an attempt to grow the business with a convenient pair of percentages: use $r\%$ of the maximum allowed recruitment and $a\%$ of the maximum allowed additional *production capacity*. There are many such combinations, but only one will achieve the highest accumulated number of delivered watches – which is what the decision-maker is responsible for. One intuitive way to decide the proportion of allocated money is to reason with the *unit productivity* of both resources: if ΔSF additional salespeople can be recruited for a *salesforce* of SF individuals, the percentage of *salesforce* increase $\Delta SF/SF$ should be the same for additional *production capacity* ΔPC as percentage of total (current) *production capacity* PC :

$$\Delta PC / PC = \Delta SF / SF. \quad (9)$$

The result of this simple calculation can then be transformed in the amount of money needed to order ΔSF and the amount needed for ΔPC . Following this idea with a *policy implementation fraction* of 0.7 (which minimizes overcapacity without reducing deliveries) in the simulation yields similar outcomes as policy *P1*, as displayed in Figure 10.

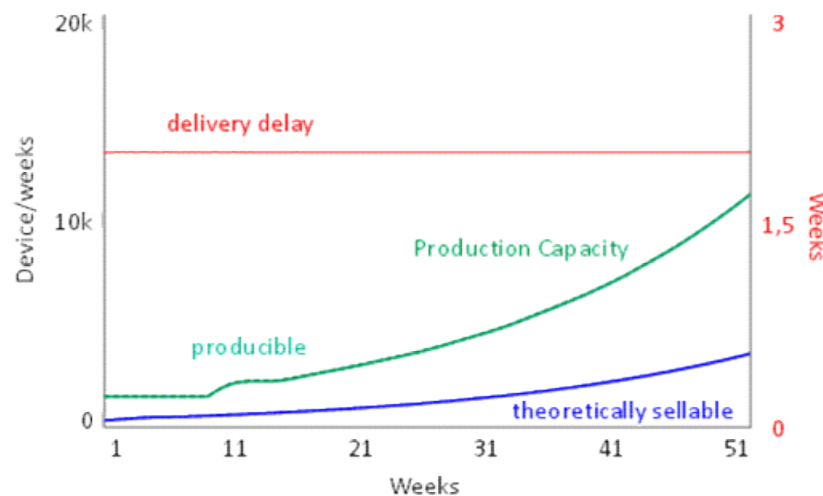


Figure 10: The behavior of the resources under policy *P3*.

Figure 10 illustrates how resources grow exponentially with a moderate curvature or acceleration, reaching a final score of 398 salespeople selling up to 3,585 watches at a *production capacity* of 6,200. At the end of the year, there is an overcapacity of 71%, meaning that for the same total quantity of deliveries – 66,284 – and the same monthly growth rate of 9%, the company has spent more financial resources than policy *P1* without need. This large overcapacity in the last week is a warning: even though over the year, the accumulated overcapacity is identical to the one observed in policy *P1* (68,723), the tendency is clearly “increasing”, and the following year ought to be viewed with worry. Therefore, while policy *P3* is sustainable, the unnecessary spending goes against the company’s goals and the decision-maker’s mission.

Regarding the decision streams of policy *P3*, Figure 11 shows that there is an initial dip followed by a trend of moderate exponential growth with permanent week-to-week. This happens since *production capacity* needs six to seven more weeks than *salesforce* to react to the decisions made each week. After the initial weeks, this difference of time de-

lays is no longer visible in the curve, even though each increase of *production capacity* in the upper graph is the consequence of a decision taken six to seven weeks earlier.

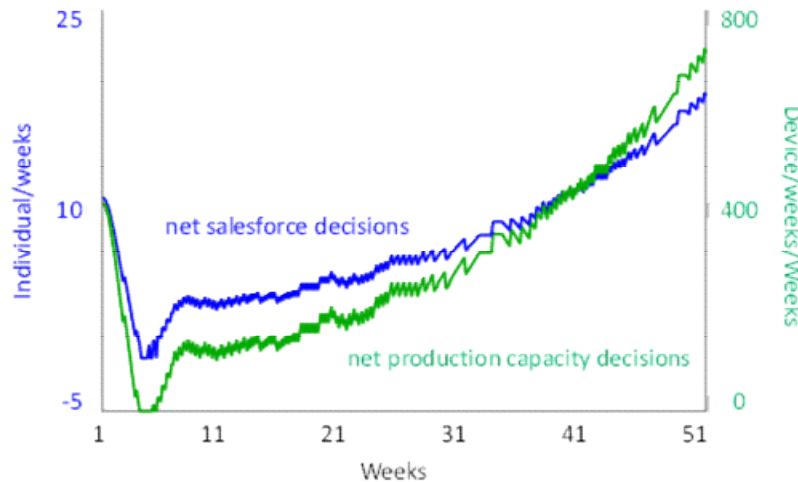


Figure 11: Adjusting the *salesforce* and the *production capacity* under policy *P3*.

4. Discussion

The following Table 2 summarizes key features of these three policies, so they can serve as benchmark for the behaviors and results of the participants in the experiment.

The *policy implementation fraction* for each of the three policies have been chosen such as to minimize the overcapacity that accumulates over the year; any other value leads to higher *production capacity* without increasing the total quantity of watches delivered. Policies *P1* and *P3* drive the *salesforce* identically and yield the same result for accumulated deliveries; however, *P3* performs slightly worse in terms of overcapacity, which is equivalent to reduced financial results. However, our primary criterion for assessing the qualities of these policies is the ability to generate sustained (exponential) growth, and the financial results only make a difference if two policies perform equally well in the primary criterion.

The results achieved by policy *P2* are huge in comparison to the other policies, with a monthly growth rate four times as high. Even if such a difference can hardly be reached in real life, the important point is that the logic behind policy *P2* leads to much higher results.

All three policies allow for smooth and exponential growth, even though *P2* is much stronger, and the behavior pattern of both resources could be sustained beyond the end of the year: even *P1* and *P2*, which maintain an overcapacity, can be defended arguing that the relative weight of the overcapacity decreases over time.

Table 2: Key features of the three stylized policies

Variables	Policy		
	<i>P1</i>	<i>P2</i>	<i>P3</i>
<i>Policy implementation fraction</i>	0.2	0.7	0.6
<i>Accumulated deliveries</i>	66,284	823,596	66,284
<i>Additional deliveries</i>	49,004	806,316	49,004

<i>Salesforce</i>	398	12,656	398
<i>Theoretically sellable</i>	3,585	113,907	3,585
<i>Producible</i>	5,930	118,860	6,200
<i>Overcapacity</i> at end of the year	2,344	4,952	2,615
<i>Overcapacity</i> at end of the year (%)	65%	4%	73%
<i>Accumulated overcapacity</i>	68,859	65,959	68,723
General features			
Total growth in %	284%	4666%	284%
Monthly growth rate	9%	38%	9%
Behavior shape	Exponential growth	Exponential growth	Exponential growth
Behavior smoothness	Continuous	Continuous	Continuous
Strength of growth	Moderate	Extreme	Moderate
Sustainable	Yes	Yes	Yes

Whereas the behavior seen in the “market growth” case showed an increase of the *delivery delay* to almost twice the value of the *desired delivery delay* – even without the “eroding goals” feedback loop and without a diminished investment in *production capacity* (see Figure 6 above) – all three policies maintain the *delivery delay* equal to the *desired delivery delay*. This implies that even policy *P1*, which follows the same line of reasoning as the “market growth” case, can be operationally specified such that *production capacity* is adjusted quickly enough to avoid overselling – which would drive the *delivery delay* up. By consequence, policy *P1* is not necessarily an idea leading to unsustainable growth followed by decline.

A second difference in terms of *delivery delay* is that the three policies avoid the oscillations observed in the “market growth” case. The fact that production capacity develops through two stages, represented as two sequential stocks, means that policy *P1* contains a second order balancing feedback loop. When decision-makers overlook the delay implied by one of these stocks – in this case *capacity under construction* – they are lead to take exaggerated decisions which later have to be compensated by further corrections in the opposite direction, like in the classical “beer game” [14, 15]. Therefore, since policy *P1* does not generate oscillations in the operationalization simulated here, it is not necessarily a decision rule that disregards the *delay*.

Since all three policies create a growth itinerary of at least 8% per period of time, and since the growth rate produced by policy *P2* is not plausible to happen in real business over sustained periods of time, the results of these simulations suggest that none of the three possibilities can be by default discarded as wrong if decision-makers do not expect or desire a higher growth rate. Remembering that there may be reasons why decision-makers in small companies are reticent towards the risks implied by policy *P2*, the results discussed here suggest that their reticence does not necessarily lead to growth-related problems, especially when the growth goal is moderate.

However, this conclusion can only be made for operational formulation of the policies that are based on the generic reasoning concerning beneath the generic policies with respect to the interdependency between the two resources *salesforce* and *production capacity*, or any other resources used to incite *demand* and to sustain *demand*:

1. An increased *salesforce* generates stress on *production capacity* and calls for a proportional effort to increase *production capacity*. The more one increases the *salesforce* per period of time, the harder it becomes to adjust the *production capacity*.

2. An increased *production capacity* generates stress on the *salesforce* and calls for a proportional increase of the *salesforce*. The more one increases the *production capacity* per period of time, the harder it becomes to adjust the *salesforce*. An alternative interpretation is: the more one increases the *production capacity* per period of time, the easier it becomes that an adequately adjusted *salesforce* can generate the *new orders* needed for the *sales revenue* to lead to a sufficient *salesforce salary budget* to sustain this increased *salesforce*.

Based on this theoretical result, some questions arise. The general reasoning can be expected from decision-makers with sufficient relevant knowledge and time to (a) develop a mental model containing the relevant elements of the situation's structure [37], to (b) infer the dynamic implications and to (c) convert them into decision rules. Clearly, not all individuals are trained or innate systems thinkers [38]. And as clearly, the time available for deliberation and policymaking is often insufficient. Therefore, it is not surprising that experimental studies have reported evidence that human individuals frequently fail to achieve sustained growth [12, 13, 39, 40]. However, the empirical finding that real people tend to take decisions that fail to achieve sustained growth calls for an explanation in terms of (a) what real individuals' mental models contain, (b) which implications they derive from their mental models and (c) what their decision rules are.

The topic of mental models can be approached in different manners. One can take the observed behaviors and externally reconstruct decision rules which are able to replicate the behaviors; this approach is compatible with the "talk is cheap" lemma of economic experiments [41] and is applied in some mental model studies [39]. However, the actual mental model of an individual is not necessarily replicated by an externally reconstructed decision-rule: even in the generic case discussed here, different policies and decision-rules like *P1* and *P3* can lead to very similar behaviors and final results. If different policies can be the driver beneath a certain behavior, the ability to generate this behavior with one policy only proves that one has found one policy to yield this behavior, but not *the* policy behind the behavior. Seen from this angle, eliciting the mental models people can articulate is a feasible way to obtain data concerning the structural representation and the reasoning of individuals, as practiced in cognitive psychology [42-45], in organization studies [46, 47] and systems research [48, 49]. A particularly relevant question follows from the fact that "mental models of dynamic systems" [37] only describe the *structure* an individual recognizes, whereas "mental models of possibilities" as used in cognitive psychology [50] represent the *reasoning*, but traditionally related to events rather than dynamic behaviors: so, can both types of mental models be combined such as to study mentally represented structure as well as the reasoning? Theoretical and empirical studies dealing with this question are called for.

For the dynamic decision situation, where interdependent resources must be steered such as to achieve sustained growth, there are several empirical questions:

1. Which decision-rules operationalize individuals' policies, what are the reasoning steps (mental models of possibilities) and which elements are in the "mental models of dynamic systems" – in particular: do they account for the interdependency and its delay?
2. Do the policies of real individuals always match with one of the three generic policies or does the set of generic policies need to be extended?
3. With which relative frequency is each generic policy chosen?
4. Who successful are the policies?

Many more specific questions will be relevant according to different disciplines like cognitive psychology, learning science, educational research, or decision science and human-computer interaction. Each particular experiment will confront participants with a large number of features, which may be more or less salient to different participants according to each participant's prior knowledge, individual personality traits and the

dynamic unfolding of the situation over the iterating decisions. For instance, assuming that the reasoning concerning the interdependencies and the stress put on one resource by the behavior of the other resource is not obvious to naïve individuals, such individuals might be cued to noticing the mutual reactions of change and stress by including a display that is made more salient when the stress on one resource increases. The salience of “stress” might depend on how different senses are provided stimuli – possibly combining vision with audition or the proprioception of movement.

There is a wide array of possibilities, and provided the design details of each experiment, the elicited data and the methods for analyzing it are made transparent, study results will be cumulative.

5. Conclusion

This article has examined a dynamical problem requiring decision-makers to achieve sustained market growth by increasing two interdependent organizational resources which react with different speeds. Since one of these resources is needed to increase demand and the other one is necessary for sustaining demand, decision-makers must choose between three general policies. Each of these policies implies a different system structure of three interlocking feedback loops:

1. Use the quick-to-react resource to drive growth (reinforcing loop) and adjust the slow-to-react resource as needed (balancing loop).
2. Use the slow-to-react resource to drive growth (reinforcing loop) and adjust the quick-to-react resource as needed (balancing loop).
3. Use both resources simultaneously to drive growth (two reinforcing loops).

Previous publications have focused on the first two policies and warned that the first one risks provoking a quick and strong surge of demand, but after this initial boom, demand cannot be satisfied and growth switches to decline and even collapse. This has been associated to the different speeds of reaction of the two resources, arguing that the delay leads decision-makers to overemphasize the quick-to-react resource. However, the simulations discussed in this article show that all three policies can in principle take the difference in speed of reaction into account: changes to one resource lay stress on the other resource, and this stress can be used to decide how much effort to put in changing the other resource. Decision-makers wishing to avoid the risks inherent in the investments required by the second policy are not condemned to overshoot and collapse. However, the second policy leads to much higher performance in terms of sales growth.

Two important limitations of this theoretical examination stem from the fact that it is based on simulation of one specific way to spell out the generic policies. Only empirical work with human decision-makers can show which of these generic policies is chosen more frequently and which specific reasoning and decision rules are put forward by these individuals. These limitations notwithstanding, the results presented here and the decision situation as implemented in the simulation model are now available for designing and carrying out experimental work.

Funding: This research received no external funding

Data Availability Statement: The simulation model can be freely used, and data can be downloaded at <https://exchange.iseesystems.com/public/martin-schaffernicht/zeiteisen-policies-explorer>.

Acknowledgments: This research has been possible due to a sabbatical (RU 753/2019) at the Technical University of Dresden, hosted by Prof. Dr. Baerbel Fuerstenau.

Conflicts of Interest: The author declares no conflict of interest.

Appendix A: Simulation model on the Internet

The model can be run on the Internet at the following URL: <https://exchange.iseesystems.com/public/martin-schaffernicht/zeiteisen-policies-explorer>

There is only one screen, as shown in the following figure:

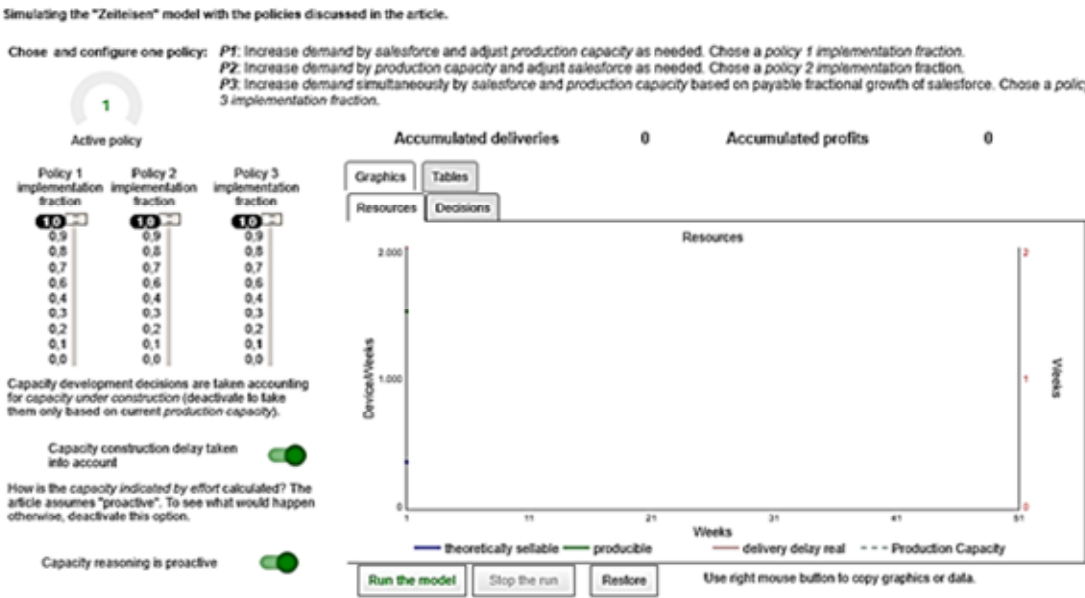


Figure A1: Screen of the user interface of the simulator

Users can choose which policy to run and set the following parameters:

- *Policy implementation fraction*: the computed value for the decisions can be used as suggested by the model or at a fraction between 0.1 and 0.9. The main text of the article discusses the behavior assuming one specific *policy implementation fraction*, but model users can explore how behavior and performance change when a different value is used.
- *Capacity construction delay accounted for*: by default, the computations are based on the total capacity, including the capacity under construction, to avoid the oscillations typical for second order negative feedback loops. This can be deactivated to explore how behaviors change.
- *Proactive capacity reasoning*: by default, the computations assuming that the degree of change of production capacity is determined proactively. This can be deactivated to explore how behaviors change.

The green "run the model" command button starts an interactive simulation, which then can be stopped by the red "stop the run" button. The "restore" button will reset all parameters to their respective default values and erase the graphics.

Appendix B: Model documentation

The simulation model has been developed using the STELLA Architect software package. The equations are presented using the following typographical conventions. **Stock** variables are printed in boldface, the connected *flow variables* are boldface italic.

All other elements are referred to as intermediate variables, including constants. The model is organized in sectors – groupings of variables – and one module (like a sub-model). Inside each sector (and module), the sequence of presentation is stocks, flows and then intermediate variables in alphabetical order. In total, the model has 8 stocks and 12 flows.

B.1. Main model

B.1.1. Salesforce

Searching(t) =

Searching(t - dt) + (to_recruit - recruited) * dt

INIT Searching = 0

UNITS: Individual

DEF: Number of currently searched for salespeople

USED BY: recruited, relative_SF_gap, total_salesforce

INFLOW: *to_recruit* =

IF(autopilot_switch=0)

THEN(now_hiring/time_unit)

ELSE(recruitment_by_policy/time_unit)

UNITS: Individual/Weeks

DEF: Weekly decision to search for additional salespeople.

USED BY: net_salesforce_decisions, Searching

OUTFLOW: *recruited* =

INT(Searching/hiring_delay)

UNITS: Individual/Weeks

DEF: Weekly actual recruitment of new salespeople.

USED BY: Searching, Sales_Force

Sales_Force(t) =

Sales_Force(t - dt) + (recruited - laid_off) * dt

INIT Sales_Force = 40

UNITS: Individual

DEF: Currently working salespeople.

USED BY: payable_additional_salesforce, orders, salesman_productivity_reported, minimum_delivery_to_pay_current_salesforce, Accounting, relative_SF_gap, expected_sales, total_salesforce, theoretically_sellable, Accounting.salesforce_costs

INFLOW: *recruited* =

INT(Searching/hiring_delay)

UNITS: Individual/Weeks

DEF: Weekly actual recruitment of new salespeople.

USED BY: Searching, Sales_Force

OUTFLOW: *laid_off* =

-now_firing/firing_delay

UNITS: Individual/Weeks

DEF: Weekly number of laid off salespeople

USED BY: net_salesforce_decisions, Sales_Force

B.1.2. Production capacity

Capacity_in_preparation(t) =

Capacity_in_preparation(t - dt) + (additional_production_capacity - going_into_service) * dt

INIT Capacity_in_preparation = 0

UNITS: Device/Weeks

DEF: Capacity units being prepared currently

USED BY: Production_Capacity_considered

INFLOW: *additional_production_capacity* = *additional_production_capacity_Participant**(*autopilot_switch*=0)+
*capacity_adjustment_by_P1**(*autopilot_switch***Policy*=1)+
*capacity_adjustment_by_P2**(*autopilot_switch***Policy*=2)+
*capacity_adjustment_by_P3**(*autopilot_switch***Policy*=3)

UNITS: Device/Weeks/Weeks

DEF: Additional Capacity units whose preparation is decided per week. If autopilot is off, then the box/slider put a fraction of the investment indicated by effort. If the autopilot is on, then the Policy controls the decision: (1) P1 and P2: investment indicated by effort; (3): P3, additional production capacity P3

USED BY: *going_into_service*, Accounting, *net_production_capacity_decisions*, Accounting, *investment_costs*, *Capacity_in_preparation*

OUTFLOW: *going_into_service* =

DELAY(additional_production_capacity; capacity_installment_delay; 0)

UNITS: Device/Weeks/Weeks

DEF: Capacity units going on-line per week

USED BY: *Capacity_in_preparation*, *Production_Capacity*

Production_Capacity(t) =

Production_Capacity(t - dt) + (going_into_service - capacity_sold_off) * dt

INIT *Production_Capacity* = *PC_init*

UNITS: Device/Weeks

DEF: Weekly number of watches that can be produced at cull capacity utilization

USED BY: *producible*, Potentials, *Production_Capacity_considered*, Accounting, *CUF_indicated*, *delivery_delay_min*, Potentials.PPC_5, Potentials.PPC_6, Potentials.PPC_7, Potentials.PPC_10, Potentials.PPC_12, Potentials.PPC_14, Potentials.PPC_15, Potentials.PPC_18, Potentials.PPC_21, Accounting, *capacity_maintenance_costs*

INFLOW: *going_into_service* =

DELAY(additional_production_capacity; capacity_installment_delay; 0)

UNITS: Device/Weeks/Weeks

DEF: Capacity units going on-line per week

USED BY: *Capacity_in_preparation*, *Production_Capacity*

OUTFLOW: *capacity_sold_off* = *-disinvestment*

UNITS: Device/Weeks/Weeks

DEF: Capacity units sold off per week

USED BY: *net_production_capacity_decisions*, *Production_Capacity*

B.1.3. Orders

Backlog(t) =

Backlog(t - dt) + (orders - deliveries) * dt

INIT *Backlog* = 720

UNITS: Device

DEF: All orders currently waiting to be produced and delivered.

USED BY: *delivery_delay_real*, *deliveries_max*, Potentials, *CUF_indicated*, *delivery_delay_min*, *desirable_deliveries*,

INFLOW: *orders* =

*INT(Sales_Force*effect_of_delivery_delay_on_demand*salesman_productivity_max)*

UNITS: Device/Weeks

DEF: Weekly number of new orders

USED BY: *salesman_productivity_reported*, *Backlog*

OUTFLOW: *deliveries* =

MIN(producible; deliveries_limit)

UNITS: Device/Weeks

DEF: Weekly number of delivered orders.

USED BY: delivery_delay_real, sales_revenues, Accounting, additional_deliveries, Accounting.variable_costs, Accounting.unit_production_cost, Accounting.variable_capital_costs, Accounting.variable_worker_costs, Backlog

B.1.4. Intermediate variables

additional_production_capacity_Participant =

IF(Capacity_adjustment_box=0)

THEN(investment_indicated_by_effort*Capacity_adjustment_slider)

ELSE(investment_indicated_by_effort*Capacity_adjustment_box)

UNITS: Device/Weeks/Weeks

DEF: Additional production capacity according to a player

USED BY: additional_production_capacity

additional_salesforce_for_excess_productibility =

theoretic_excess_productibility/(0,9*salesman_productivity_max)

UNITS: Individual

DEF: Number of additional salespeople that would be needed to sell those watches that can be produced but not sold by the current workforce.

USED BY: recruitment_by_policy, Salesforce_fractional_effort_from_Capacity

additional_SF_monthly_salary_costs =

payable_additional_salesforce*salesperson_monthly_salary

UNITS: Euros/Month

DEF: Money implied by salaries of the additional salespeople.

autopilot_switch = 1

UNITS: unitless

DEF: if set to 1, the model policies regulate the decisions; if set to 0, users take the decisions.

USED BY: deliveries_limit, additional_production_capacity, to_recruit

capacity_adjustment_indicated_by_effort =

Capacity_indicated_by_effort-Production_Capacity_considered

UNITS: Device/Weeks

DEF: Theoretically indicated additional production capacity units according to the "stress" reasoning. This is the stock's value, so it cannot directly be used in a flow.

USED BY: investment_indicated_by_effort, disinvestment_indicated_by_effort

Capacity_fractional_effort =

(desirable_deliveries/theoretically_productible)/2

UNITS: unitless

USED BY: Capacity_indicated_by_effort

Capacity_adjustment_box = 1

UNITS: unitless

DEF: Value of the edit box in the user interface

USED BY: additional_production_capacity_Participant, disinvestment

Capacity_adjustment_slider = 0

UNITS: unitless

DEF: Value of the slider in the user interface

USED BY: additional_production_capacity_Participant, disinvestment

Capacity_indicated_by_effort = Produc-

tion_Capacity_considered*(proactiveness+Capacity_fractional_effort)

UNITS: Device/Weeks

USED BY: capacity_adjustment_indicated_by_effort

capacity_installment_delay = 8

UNITS: Weeks

USED BY: going_into_service

Capacity_Utilization_Fraction = Shifts_per_week/Shifts_per_week_max

UNITS: unitless
USED BY: producible, theoretically_productible

Days_per_week = 7
UNITS: days/week
USED BY: Shifts_per_week

delay_perceived = 1
UNITS: unitless
DEF: Allows to simulate according to if the decision-maker takes the capacity installment delay into account (=1, default) or not (=0: like in the literature)
USED BY: Production_Capacity_considered

delay_perception_time = 6
UNITS: Weeks
USED BY: delivery_delay_perceived

deliveries_limit =
INT(deliveries_max/(1+autopilot_switch))
UNITS: Device/Weeks
USED BY: deliveries

delivery_delay_perceived =
SMTH1(delivery_delay_real; delay_perception_time; delivery_delay_real)
UNITS: Weeks
DEF: The clients' expectation regarding the delivery delay, which adjusts progressively to the actual delivery delay.
USED BY: effect_of_delivery_delay_on_demand

delivery_delay_real =
SAFEDIV(Backlog; deliveries)
UNITS: Weeks
DEF: Actual number of weeks it takes between order and delivery
USED BY: delivery_delay_perceived, reported_delivery_delay

disinvestment =
IF(Capacity_adjustment_box=0)
THEN(disinvestment_indicated_by_effort*Capacity_adjustment_slider)
ELSE(disinvestment_indicated_by_effort*Capacity_adjustment_box)
UNITS: Device/Weeks/Weeks
USED BY: capacity_sold_off

disinvestment_indicated_by_effort =
MIN(0; capacity_adjustment_indicated_by_effort)/time_unit
UNITS: Device/Weeks/Weeks
DEF: Theoretically indicated excess production capacity units according to the "stress" reasoning. Adjusted to time units to fit as flow variable.
USED BY: disinvestment

effect_of_delivery_delay_on_demand = GRAPH(delivery_delay_perceived)

Input	Output
0	1.000
1	0.970
2	0.900
3	0.730
4	0.530
5	0.380
6	0.250
7	0.150
8	0.080
9	0.030
10	0.020

UNITS: unitless
 DEF: Effect of the expectation regarding the delivery delay on the willingness to buy.
 USED BY: orders

expected_sales_revenues =
 SMTH1(sales_revenues; revenue_averaging_time; sales_revenues)
 UNITS: Euros/Weeks
 DEF: The expected sales revenues are a moving average of the past weeks
 USED BY: payable_salesforce, budget_for_salesforce_costs

firing_delay = 1
 UNITS: Weeks
 DEF: Time it takes to implement layoff decisions
 USED BY: laid_off

hiring_delay =
 1+relative_SF_gap*(hiring_delay_max-hiring_delay_min)
 UNITS: Weeks
 DEF: Time it takes to find and recruit the number of additional salespeople sought for. Depends on the relationship of the number of searched salespeople to the current salesforce.
 USED BY: recruited

hiring_delay_max = 4
 UNITS: Weeks
 DEF: Longest possible time to find and recruit additional salespeople.
 USED BY: hiring_delay

hiring_delay_min = 1
 UNITS: Weeks
 DEF: Shortest possible time to find and recruit additional salespeople.
 USED BY: hiring_delay

investment_indicated_by_effort =
 MAX(capacity_adjustment_indicated_by_effort; 0)/time_unit
 UNITS: Device/Weeks/Weeks
 DEF: Theoretically indicated additional production capacity units according to the "stress" reasoning. Adjusted to time units to fit as flow variable.
 USED BY: additional_production_capacity_Participant, capacity_adjustment_by_P1, capacity_adjustment_by_P2, additional_capacity_investment_costs

Sales_force_box = 1
 UNITS: unitless
 DEF: Value of the edit box in the user interface
 USED BY: now_hiring_C, now_firing_C

Sales_force_slider = 0
 UNITS: unitless
 DEF: Value of the slider in the user interface
 USED BY: now_hiring_E, now_firing_E

recruitment_by_policy =
 IF(Policy=2) THEN(additional_salesforce_for_excess_producibility)
 ELSE(payable_additional_salesforce)
 UNITS: Individual
 DEF: Weekly number of salespeople to be searched and recruited according to one of the policies implemented in the model. If it is policy P2, then hiring is driven by the additional salesforce for excess producibility, otherwise it is driven by payable additional salesforce.
 USED BY: to_recruit

```

now_firing =
IF(now_firing_C=0) THEN(now_firing_E) ELSE(now_firing_C)
    UNITS: Individual
    USED BY: laid_off
now_firing_C =
INT(Sales_force_box*MIN(0; payable_additional_salesforce))
    UNITS: Individual
    USED BY: now_firing
now_firing_E =
INT(Sales_force_slider*MIN(0; payable_additional_salesforce))
    UNITS: Individual
    USED BY: now_firing
now_hiring =
IF(now_hiring_C=0) THEN(now_hiring_E) ELSE(now_hiring_C)
    UNITS: Individual
    DEF: Weekly number of salespeople to be searched and recruited according to a
player
    USED BY: to_recruit
now_hiring_C =
INT(Sales_force_box*MAX(payable_additional_salesforce; 0))
    UNITS: Individual
    DEF: Weekly number of salespeople to be searched and recruited according to a
player if the user interface has an edit box as control.
    USED BY: now_hiring
now_hiring_E =
INT(Sales_force_slider*MAX(payable_additional_salesforce; 0))
    UNITS: Individual
    DEF: Weekly number of salespeople to be searched and recruited according to a
player if the user interface has a slider as control.
    USED BY: now_hiring
payable_additional_salesforce =
payable_salesforce-Sales_Force
    UNITS: Individual
    DEF: Number of additional salespeople which can be paid, accounting for the
currently searched salespeople (to avoid oscillations).
    USED BY: recruitment_by_policy, now_hiring_E, now_hiring_C, now_firing_C,
now_firing_E, additional_SF_monthly_salary_costs, sales_force_growth_rate,
Salesforce_fractional_effort_from_budget
payable_salesforce =
INT((revenue_fraction_for_salesforce_salaries *expected_sales_revenues)
/(salesperson_monthly_salary/weeks_per_month))
    UNITS: Individual
    DEF: Payable number of salespeople according to the salaries and the expected sales
revenues, assuming a stable fraction
    USED BY: payable_additional_salesforce, Potentials, Potentials.payable_sales
PC_init = 1512
    UNITS: Device/Weeks
    DEF: 1512
proactiveness = 1
    UNITS: unitless
    DEF: Used to determine capacity indicated by effort, which can be in a reactive
mode to replicate results from Forrester and from Morecroft, or in a proactive mode to
drive, for instance, the weightlifter.
    USED BY: Capacity_indicated_by_effort

```


producible =
 INT(Capacity_Utilization_Fraction*Production_Capacity)
 UNITS: Device/Weeks
 USED BY: deliveries
 Production_Capacity_considered = de-
 lay_perceived*Capacity_in_preparation+Production_Capacity
 UNITS: Device/Weeks
 USED BY: theoretically_productible, Capacity_indicated_by_effort, capaci-
 ty_adjustment_indicated_by_effort, capacity_adjustment_by_P3, Capaci-
 ty_indicated_by_delay_gap, capacity_adjustment_indicated_by_delay_gap
 relative_SF_gap = Searching/Sales_Force
 UNITS: unitless
 DEF: The relationship of the number of searched salespeople to the current
 salesforce.
 USED BY: hiring_delay
 revenue_averaging_time = 3
 UNITS: Weeks
 DEF: Number of weeks in the moving average for sales values expected
 USED BY: expected_sales_revenues
 revenue_fraction_for_salesforce_salaries = 0,25
 UNITS: unitless
 DEF: Percentage of the expected sales revenues which can be used for paying
 salesforce salaries. In the original mode = 0,1. Under the price and salary assumptions of
 the revised model, it has to be at least 19.5% (initialization equilibrium). If = 25%, the
 default situation generates the possibility to pay 51 salespeople, therefore there is space
 to hire additional salesforce.
 USED BY: payable_salesforce, budget_for_salesforce_costs
 sales_revenues =
 deliveries*unit_sales_price
 UNITS: Euros/Weeks
 DEF: Amount received for the watches delivered weekly
 USED BY: expected_sales_revenues, Accounting, Accounting.total_revenues
 salesman_productivity_max = 10
 UNITS: Device/Weeks/Individual
 DEF: Highest theoretical number of weekly new orders one salesperson can achieve.
 USED BY: orders, minimum_delivery_to_pay_current_salesforce, expected_sales,
 salesforce_costs_per_watch, theoretically_sellable, addition-
 al_salesforce_for_excess_productibility
 salesperson_monthly_salary = 3000
 UNITS: Euros/Month/Individual
 DEF: Monthly salary for each of the salespeople.
 USED BY: payable_salesforce, Accounting, additional_SF_monthly_salary_costs,
 salesforce_costs_per_watch, Accounting.salesforce_costs
 Shifts_per_day = 3
 UNITS: shifts/days
 USED BY: Shifts_per_week
 Shifts_per_week =
 Days_per_week*Shifts_per_day
 UNITS: shifts/week
 USED BY: Capacity_Utilization_Fraction, Accounting, Account-
 ing.extra_shifts_multiplier
 Shifts_per_week_max = 21
 UNITS: shifts/week
 USED BY: Capacity_Utilization_Fraction

$\text{theoretic_excess_productibility} =$
 $\text{MAX}(\text{theoretically_productible} - \text{theoretically_sellable}; 0)$
 UNITS: Device/Weeks
 DEF: Weekly number of producible watches that would not be sold with the current salesforce.
 USED BY: $\text{additional_salesforce_for_excess_producibility}$
 $\text{theoretically_productible} =$
 $\text{Capacity_Utilization_Fraction} * \text{Production_Capacity_considered}$
 UNITS: Device/Weeks
 DEF: Weekly number of producible watches according to the considered production capacity and the capacity utilization fraction.
 USED BY: $\text{Capacity_fractional_effort}$, $\text{theoretic_excess_productibility}$
 $\text{theoretically_sellable} =$
 $\text{Sales_Force} * 0.9 * \text{salesman_productivity_max}$
 UNITS: Device/Weeks
 DEF: Weekly number of new orders which the salesforce could achieve at the maximum productivity
 USED BY: $\text{theoretic_excess_productibility}$
 $\text{unit_sales_price} = 431$
 UNITS: Euros/Device
 DEF: Amount paid by clients per watch
 USED BY: sales_revenues
 $\text{weeks_per_month} = 4$
 UNITS: Weeks/Month
 DEF: Each month is assumed to last exactly 4 weeks
 USED BY: $\text{payable_salesforce}$, Accounting , $\text{salesforce_costs_per_watch}$, $\text{Accounting_salesforce_costs}$

B.2. Policy P1

$\text{capacity_adjustment_by_P1} = (\text{invest-}$
 $\text{ment_indicated_by_effort} * \text{Policy_1_by_effort} + \text{capacity_adjustment_indicated_by_delay}$
 $\text{_gap} * (1 - \text{Policy_1_by_effort})) * \text{P1_fraction}$
 UNITS: Device/Weeks/Weeks
 USED BY: $\text{additional_production_capacity}$
 $\text{capacity_adjustment_indicated_by_delay_gap} =$
 $(\text{Capacity_indicated_by_delay_gap} - \text{Production_Capacity_considered}) / \text{time_unit}$
 UNITS: Device/Weeks/Weeks
 USED BY: $\text{capacity_adjustment_by_P1}$
 $\text{Capacity_indicated_by_delay_gap} =$
 $\text{delivery_delay_gap_relative} * \text{Production_Capacity_considered}$
 UNITS: Device/Weeks
 USED BY: $\text{capacity_adjustment_indicated_by_delay_gap}$
 $\text{delivery_delay_gap} =$
 $\text{reported_delivery_delay} - \text{desired_delivery_delay}$
 UNITS: Weeks
 USED BY: $\text{delivery_delay_gap_relative}$
 $\text{delivery_delay_gap_relative} =$
 $\text{delivery_delay_gap} / \text{desired_delivery_delay}$
 UNITS: unitless
 USED BY: $\text{Capacity_indicated_by_delay_gap}$
 $\text{delivery_delay_reporting_time} = 4$
 UNITS: Weeks
 USED BY: $\text{reported_delivery_delay}$

$P1_fraction = 1$
 UNITS: unitless
 USED BY: capacity_adjustment_by_P1
 $Policy_1_by_effort = 1$
 UNITS: unitless
 DEF: 1: depending on the effort logic
 0: reactive, depending on the delay gap
 USED BY: capacity_adjustment_by_P1
 $reported_delivery_delay =$
 $SMTH1(delivery_delay_real; delivery_delay_reporting_time; delivery_delay_real)$
 UNITS: Weeks
 USED BY: DD_underrun, delivery_delay_gap

B.3. Policy P2

$capacity_adjustment_by_P2 =$
 $P2_fraction * investment_indicated_by_effort$
 UNITS: Device/Weeks/Weeks
 USED BY: additional_production_capacity
 $P2_fraction = 1$
 UNITS: unitless
 USED BY: capacity_adjustment_by_P2

B.4. Policy P3

$capacity_adjustment_by_P3 =$
 $(sales_force_growth_rate * P3a_fraction) * Production_Capacity_considered / time_unit$
 UNITS: Device/Weeks/Weeks
 DEF: In salesforce changes X%, so shall the production capacity.
 USED BY: additional_production_capacity
 $P3a_fraction = 1$
 UNITS: unitless
 USED BY: capacity_adjustment_by_P3
 $sales_force_growth_rate =$
 $payable_additional_salesforce / total_salesforce$
 UNITS: unitless
 USED BY: capacity_adjustment_by_P3

B.5. Stress_on_salesforce

$Salesforce_fractional_effort_from_budget =$
 $SAFEDIV(payable_additional_salesforce; total_salesforce; 0)$
 UNITS: unitless
 DEF: Fraction of the salary costs for the additional required salesforce as compared to the salary costs of the current total salesforce.
 This was one option for representing the stress on the salesforce resource, but it is currently not used.
 $Salesforce_fractional_effort_from_Capacity =$
 $SAFEDIV(additional_salesforce_for_excess_producibility; total_salesforce; 0)$
 UNITS: unitless
 DEF: Fraction of additional salespeople which would be required as compared to the current total salesforce. This is not used in the model's computations, but fed to the user (human decision-maker) on the user interface.
 $total_salesforce =$
 $Sales_Force + Searching$

UNITS: Individual
 USED BY: Salesforce_fractional_effort_from_Capacity, sales_force_growth_rate,
 Salesforce_fractional_effort_from_budget
B.6. Auxiliary_calculations
Accumulated_DD_underrun(t) =
Accumulated_DD_underrun(t - dt) + (DD_underrun) * dt
 INIT Accumulated_DD_underrun = 0
 UNITS: Weeks
 INFLOW: *DD_underrun* =
MAX(2-reported_delivery_delay; 0)/time_unit
 UNITS: unitless
 USED BY: Accumulated_DD_underrun
Accumulated_deliveries(t) =
Accumulated_deliveries(t - dt) + (additonal_deliveries) * dt
 INIT Accumulated_deliveries = 0
 UNITS: Device
 INFLOW: *additonal_deliveries* =
deliveries
 UNITS: Device/Weeks
 USED BY: Accumulated_deliveries
 additional_capacity_investment_costs = invest-
 ment_indicated_by_effort*Accounting.investment_cost_of_capacity_unit
 UNITS: Euros
 CUF_indicated =
 ((Backlog/2)/time_unit)/Production_Capacity
 UNITS: unitless
 Current_week = TIME
 UNITS: Weeks
 USED BY: Game_on
 deliveries_max =
 Backlog/time_unit
 UNITS: Device/Weeks
 USED BY: deliveries_limit
 delivery_delay_min =
 Backlog/Production_Capacity
 UNITS: Weeks
 desirable_deliveries =
 Backlog/desired_delivery_delay
 UNITS: Device/Weeks
 DEF: The weekly deliveries which are just enough to keep the delivery delay equal
 to the desired delivery delay
 USED BY: Capacity_fractional_effort
 desired_delivery_delay = 2
 UNITS: Weeks
 DEF: The goal: 2 weeks
 USED BY: desirable_deliveries, delivery_delay_gap, delivery_delay_gap_relative
 expected_sales =
 Sales_Force*salesman_productivity_max*0,9
 UNITS: Device/Weeks
 Game_on =
 51-Current_week
 UNITS: Weeks

$\text{minimum_delivery_to_pay_current_salesforce} =$
 $(\text{salesman_productivity_max} * 0.85) * \text{Sales_Force}$
 UNITS: Device/Weeks
 $\text{net_production_capacity_decisions} =$
 $\text{additional_production_capacity} - \text{capacity_sold_off}$
 UNITS: Device/Weeks/Weeks
 $\text{net_salesforce_decisions} =$
 $\text{to_recruit} - \text{laid_off}$
 UNITS: Individual/Weeks
 $\text{salesman_productivity_reported} =$
 $\text{orders} / \text{Sales_Force}$
 UNITS: Device/Weeks/Individual
 $\text{time_unit} = 1$
 UNITS: Weeks
 USED BY: CUF_indicated, deliveries_max, DD_underrun, to_recruit, disinvestment_indicated_by_effort, capacity_adjustment_by_P3, investment_indicated_by_effort, capacity_adjustment_indicated_by_delay_gap

B.7. Accounting module

Accounting.Accumulated_profits(t) =
Accumulated_profits(t - dt) + (weekly_profits) * dt
 INIT Accounting.Accumulated_profits = 0
 UNITS: Euros
 INFLOWS: *accounting.weekly_profits* =
total_revenues - total_costs
 UNITS: Euros/Weeks
 USED BY: Accounting.Accumulated_profits
 Accounting.capacity_maintenance_costs =
 .Production_Capacity * maintenance_costs_per_capacity_unit
 UNITS: Euros/Weeks
 USED BY: Accounting.total_costs, Accounting.total_production_costs
 Accounting.capacity_sold_off = 0
 UNITS: Device/Weeks/Weeks
 USED BY: Accounting.sell_off_revenue
 Accounting.effective_variable_cost_per_unit =
 $(1 - \text{labor_cost_fraction}) * \text{variable_costs_per_unit} + \text{labor_costs_per_unit}$
 UNITS: Euros/Device
 USED BY: Accounting.variable_costs
 Accounting.extra_shifts_multiplier = GRAPH(Shifts_per_week)

Input	Output
0	0.0000
1	0.0000
2	0.0000
3	0.0000
4	0.0000
5	1.0000
6	1.041666667
7	1.107142857
8	0.0000
9	0.0000

10	1.1250
11	0.0000
12	1.1666666667
13	0.0000
14	1.232142857
15	1.2500
16	0.0000
17	0.0000
18	1.291666667
19	0.0000
20	0.0000
21	1.357142857

UNITS: unitless

USED BY: Accounting.labor_costs_per_unit

Accounting.investment_cost_of_capacity_unit = 20000

UNITS: Euros/(Device/Weeks)

USED BY: Accounting.investment_costs, Accounting.sell_off_price_per_unit, additional_capacity_investment_costs, capacity_cost_per_watch, additional_production_capacity_P3_b

Accounting.investment_costs =

.additional_production_capacity*investment_cost_of_capacity_unit

UNITS: Euros/Weeks

USED BY: Accounting.total_costs

Accounting.labor_cost_fraction = 0,7

UNITS: unitless

USED BY: Accounting.labor_cost_per_unit_normal, Accounting.effective_variable_cost_per_unit, Accounting.variable_capital_costs

Accounting.labor_cost_per_unit_normal =

labor_cost_fraction*variable_costs_per_unit

UNITS: Euros/Device

USED BY: Accounting.labor_costs_per_unit

Accounting.labor_costs =

variable_worker_costs+salesforce_costs

UNITS: Euros/Weeks

Accounting.labor_costs_per_unit =

labor_cost_per_unit_normal*extra_shifts_multiplier

UNITS: Euros/Device

USED BY: Accounting.effective_variable_cost_per_unit, Accounting.variable_worker_costs

Accounting.maintenance_costs_per_capacity_unit = 0

UNITS: Euros/Device

DEF: Was originally set to 48. Put to 0 on December 4 2019 because of an unclear computation error.

USED BY: Accounting.capacity_maintenance_costs

Accounting.salesforce_costs = .

salesperson_monthly_salary/.weeks_per_month*.Sales_Force

UNITS: Euros/Weeks

USED BY: Accounting.total_costs, Accounting.labor_costs

Accounting.sell_off_discount_fraction = 0,5

UNITS: unitless

USED BY: Accounting.sell_off_price_per_unit

Accounting.sell_off_price_per_unit =

(1-sell_off_discount_fraction)*investment_cost_of_capacity_unit

UNITS: Euros/(Device/Weeks)
 USED BY: Accounting.sell_off_revenue
 Accounting.sell_off_revenue =
 capacity_sold_off*sell_off_price_per_unit
 UNITS: Euros/Weeks
 USED BY: Accounting.total_revenues
 Accounting.total_costs =
 investment_costs+capacity_maintenance_costs+variable_costs+salesforce_costs
 UNITS: Euros/Weeks
 USED BY: Accounting.weekly_profits
 Accounting.total_production_costs =
 variable_costs+capacity_maintenance_costs
 UNITS: Euros/Weeks
 USED BY: Accounting.unit_production_cost
 Accounting.total_revenues = .
 sales_revenues+sell_off_revenue
 UNITS: Euros/Weeks
 USED BY: Accounting.weekly_profits
 Accounting.unit_production_cost =
 SAFEDIV(total_production_costs; .deliveries)
 UNITS: Euros/Device
 Accounting.variable_capital_costs = .
 deliveries*((1-labor_cost_fraction)*variable_costs_per_unit)
 UNITS: Euros/Weeks
 Accounting.variable_costs = .
 deliveries*effective_variable_cost_per_unit
 UNITS: Euros/Weeks
 USED BY: Accounting.total_costs, Accounting.total_production_costs
 Accounting.variable_costs_per_unit = 200
 UNITS: Euros/Device
 USED BY: Accounting.labor_cost_per_unit_normal, Accounting.effective_variable_cost_per_unit, Accounting.variable_capital_costs
 Accounting.variable_worker_costs = .
 deliveries*labor_costs_per_unit
 UNITS: Euros/Weeks
 USED BY: Accounting.labor_costs

6. References

1. Barney J. Firm Resources and Sustained Competitive Advantage. *Journal of Management*. 1991;17(1):22.
2. Crook TR, Ketchen DJ, Combs JG, Todd SY. Strategic resources and performance: a meta-analysis. *Strategic Management Journal*. 2008;29(11):1141-54.
3. Kor YY, Mahoney JT. Edith Penrose's (1959) Contributions to the Resource-based View of Strategic Management. *Journal of Management Studies*. 2004;41(1):11.
4. Penrose ET. *The Theory of the Growth of the Firm*. New York: John Wiley; 1959.
5. Peteraf MA. The Cornerstones of Competitive Advantage: A Resource-Based View. *Strategic Management Journal*. 1993;14(3):12.
6. Wernerfelt B. A Resource-Based view of the Firm. *Strategic Management Journal*. 1984;5(2):10.

7. Warren K. Improving strategic management with the fundamental principles of system dynamics. *System Dynamics Review*. 2005;21(4):329-50.
8. Barney J. Is the Resource-Based "View" a Useful Perspective for Strategic Management Research? Yes. *Academy of Management Review*. 2001;26(1):16.
9. Kunc MH, Morecroft JDW. Managerial decision making and firm performance under a resource-based paradigm. *Strategic Management Journal*. 2010;31(11):1164-82.
10. Priem RL, Butler JE. Is the Resource-Based "View" a Useful Perspective for Strategic Management Research? *Academy of Management Review*. 2001;26(1):19.
11. Özgün O, Barlas Y. Effects of systemic complexity factors on task difficulty in a stock management game. *System Dynamics Review*. 2015;31(3):115-46.
12. Moxnes E. Not only the tragedy of the commons: misperceptions of feedback and policies for sustainable development. *System Dynamics Review*. 2000;16(4):325-48.
13. Moxnes E. Misperceptions of basic dynamics: the case of renewable resource management. *System Dynamics Review*. 2004;20(2):139-62.
14. Sterman J. Misperceptions of Feedback in Dynamic Decision Making. *Organizational Behavior and Human Decision Processes*. 1989;43(3):301-35.
15. Sterman J. Modeling managerial behavior - misperceptions of feedback in a dynamic decision-making experiment. *Management Science*. 1989;35:18.
16. Baghaei Lakeh A, Ghaffarzadegan N. Does analytical thinking improve understanding of accumulation? *System Dynamics Review*. 2015;31(1-2):46-65.
17. Cronin MA, Gonzalez C. Understanding the building blocks of dynamic systems. *System Dynamics Review*. 2007;23(1):1-17.
18. Cronin MA, Gonzalez C, Sterman JD. Why don't well-educated adults understand accumulation? A challenge to researchers, educators, and citizens. *Organizational Behavior and Human Decision Processes*. 2009;108(1):116-30.
19. Gonzalez C, Wong H-y. Understanding stocks and flows through analogy. *System Dynamics Review*. 2012;28(1):3-27.
20. Sterman J. Does formal system dynamics training improve people's understanding of accumulation? *System Dynamics Review*. 2010;26(4):316-34.
21. Stouten H, Größler A. Task Complexity in Individual Stock Control Tasks for Laboratory Experiments on Human Understanding of Dynamic Systems. *Systems Research and Behavioral Science*. 2017;34(1):62-77.
22. Sweeney LB, Sterman J. Bathtub dynamics: initial results of a systems thinking inventory. *System Dynamics Review*. 2000;16(4):249-86.
23. Paich M, Sterman J. Boom, Bust, and Failures to Learn in Experimental Markets. *Management Science*. 1993;39(12):1439-58.
24. Arquitt S, Honggang X, Johnstone R. A system dynamics analysis of boom and bust in the shrimp aquaculture industry. *System Dynamics Review*. 2005;21(4):305-24.
25. Ford A. Boom & Bust in Power Plant Construction: Lessons from the California Electricity Crisis. *Journal of Industry, Competition and Trade*. 2002;2(1.2):59-74.
26. Gary MS, Dosi G, Lovallo D. Boom and Bust Behavior: On the Persistence of Strategic Decision Biases. In: Hodgkinson G, Starbuck W, editors. *The Oxford handbook of organizational decision making*. Oxford: Oxford University Press; 2008. p. 33-55.
27. Morecroft JD, editor *System Dynamics , RBV and Behavioural Theories of Firm Performance: Lessons from People Express*. 28th International Conference of the System Dynamics Society; 2008; Athens: System Dynamics Society.

28. Warren K. Competitive Strategy Dynamics. Chichester: John Wiley; 2008.
29. Forrester JW. Market Growth as Influenced by Capital Investment. *Industrial Management Review*. 1975;9(2):23.
30. Morecroft JD. Strategic Modelling and Business Dynamics. Chichester: John Wiley; 2015.
31. Oliva R, Sterman JD, Giese M. Limits to growth in the new economy: exploring the "get big fast" strategy in e-commerce. *System Dynamics Review*. 2003;19(2):83-117.
32. Sterman JD, Henderson R, Beinhocker ED, Newman LI. Getting Big Too Fast: Strategic Dynamics with Increasing Returns and Bounded Rationality. *Management Science*. 2007;53(4):683-96.
33. Sterman JD, Paich M. Boom, Bust, and Failures to Learn in Experimental Markets. *Management Science*. 1933;39(12):20.
34. Arango S, Larsen E. Cycles in deregulated electricity markets: Empirical evidence from two decades. *Energy Policy*. 2011;39(5):9.
35. Schaffernicht MF. Causal loop diagrams as means to improve the understanding of dynamic problems: a critical analysis. *Systems Research and Behavioral Science*. 2010;27(6):13.
36. Sterman J. Business dynamics - Systems Thinking and Modelling for a Complex World: McGraw Hill; 2000.
37. Groesser SN, Schaffernicht MF. Mental models of dynamic systems: taking stock and looking ahead. *System Dynamics Review*. 2012;28(1):22.
38. Sweeney LB, Sterman J. Thinking about systems: student and teacher conceptions of natural and social systems. *System Dynamics Review*. 2007;23(2/3):285-312.
39. Gary MS, Wood RE. Unpacking mental models through laboratory experiments. *System Dynamics Review*. 2016;32(2):101-29.
40. Rahmandad H, Repenning N, Sterman J. Effects of feedback delay on learning. *System Dynamics Review*. 2009;25(4):309-38.
41. Arango A. S, Castañeda A. JA, Olaya M. Y. Laboratory experiments in the system dynamics field. *System Dynamics Review*. 2012;28(1):94-106.
42. Byrne RMJ, Johnson-Laird PN. If and or: Real and counterfactual possibilities in their truth and probability. *Journal of Experimental Psychology: Learning, Memory, and Cognition*. 2020;46(4):760-80.
43. Johnson-Laird PN. Mental Models Towards a Cognitive Science of Language. Cambridge, UK: Cambridge University Press; 1983.
44. Johnson-Laird PN, Byrne RMJ. Conditionals: A theory of meaning, pragmatics, and inference. *Psychological Review*. 2002;109(4):646-78.
45. Ragni M, Johnson-Laird P. Reasoning about epistemic possibilities. *Acta Psychologica*. 2020;208.
46. Langan-Fox J, Anglim J, Wilson JR. Mental Models, Team Mental Models, and Performance: Process, Development, and Future Directions. *Human Factors and Ergonomics in Manufacturing*. 2004;14(4):21.
47. Mohammed S, Klimoski R, Rentsch JR. The Measurement of Team Mental Models: We Have No Shared Schema. *Organizational Research Methods*. 2000;3(2):123-65.
48. Schaffernicht MF. Causal attributions of vineyard executives – A mental model study of vineyard management. *Wine Economics and Policy*. 2017;6(2):107-35.
49. Schaffernicht MF. Unrecognized interdependencies in mental models – the case for taking feedback loops into account. *Systems Research and Behavioral Science*. 2019;36(4):580-603.
50. Johnson-Laird PN. Mental models and human reasoning. *Proc Natl Acad Sci U S A*. 2010;107(43):18243-50.