2017 21st Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES)

Evolutionary programming of product design policies. An agent-based model study.

Ben Vermeulen*, Bin-Tzong Chie†, Shu-Heng Chen†, Andreas Pyka*
* Chair of Innovation Economics, Institute of Economics, University of Hohenheim, Stuttgart, Germany
Email: b.vermeulen@uni-hohenheim.de; a.pyka@uni-hohenheim.de
† Department of Industrial Economics, Tamkang University, Taipei, Taiwan
Email: bt.chie@gms.tku.edu.tw
‡ AI-Econ Research Center, National Chengchi University, Taipei, Taiwan
Email: chen.shuheng@gmail.com

***Abstract*—The new product development and operational marketing literature grapples with incorporation of uncertainty on the market and technological structure discovered over time. In contrast, market and technological uncertainty is at the heart of neo-Schumpeterian agent-based models used in evolutionary innovation economics. We present a novel agent-based model in which designer agents design products to cater to services desired by user agents. In this model, designers imitate and experiment with design policies with which they engage in contests to puzzle together products. This model thus 'evolutionary programs' a commendable design policy for the given market and technological structure. We experimentally vary the segmentation of the market and the density of technological relationships ex ante unknown to designer agents and then study the emerging 'winning' design policies. Preliminary simulation results reveal that there is no 'one-size-fits-all' design policy, but that winning design policies are tailored to the structure of market and technology following particular rationales. Given that we present a novel model, we critically reflect on the operationalizations and propose further refinements.**

I. Introduction

The motivation for the model presented in this paper comes from an observation made in a project on the design of robots. Field studies revealed that particular robots which were successfully developed and implemented in South-Korea failed to be adopted in Denmark and Finland (see e.g. [3]). Designing of a robot with only one particular target group in mind may inhibit a successful adoption by another group at a later stage. In retrospect, the designers of these robots could have better explored the market more extensively to design a robot which caters to the preferences of a bigger market.

When turning to the traditional new product development and design (NPD) literature [19], [27], it is argued that after one has established technical opportunities, one should conduct a proper market study to identify user needs, and then determine which concrete product is going to be designed to target a subset of user needs. In operational marketing literature, mathematical solvers are used to pick the technical attributes of products that maximize market returns, even from a segmented market (cf. [14]). This new product development literature is in stark contrast with the evolutionary innovation economic literature, in which it is argued that product designs are outcome of a competitive race of autonomous firms vying for customer demand, whereby firms suffer both technological and market uncertainties. The underlying assumption is that firm agents do not know whether a particular avenue of technological developments leads to feasible products and whether there is demand for the product being designed. For one, users' needs and wants can often only be articulated or are in part even created only whenever the technological options are presented to them in materialized form. Moreover, firms often direct their scarce research and development resources in technological directions based on market feedback. The last decade, this 'evolutionary perspective' focusing on uncertainty is permeating the new product development and design literature as well [21]. Indeed, product development and design processes are to be tailored to the particularities, and should be able to flexibly respond to new market and technical insights [28].

The research question now is whether and, if so, how to tailor the product design policy to the (uncertainty about) market and technological structures. In this paper, we bridge the gap between the two bodies of literature, operational marketing research on new product design and evolutionary economic perspective on innovation. For this we use an operational agent-based model in which designer agents are engaged in repeated contests in which products are designed while facing uncertainty in technological relationships and uncertainty in market demand. From one to the next design contest, poorly performing agents imitate design policies of agents with superior performance, and, as such, the agent-based model 'evolutionary programs' superior design policies producing technologically feasible and market viable designs. In this conference paper, we present our novel agent-based model and provide preliminary simulation results.

II. Literature

In the mainstream extant product development and design literature (see those reviewed in [19]), firms rationally match product attributes to customers' preferences for these attributes so as to maximize profit (see e.g. [14], [15], [17], [24]). However, this requires perfect insight in customer preferences and perfect modularity of combinations of technical characteristics, and requires selection of service characteristics and module recombination at the same time (see [11]). However, insights

978-1-5386-0743-5/17/$31.00

in requirements and desires of customers is often limited (in part due to lack of customer research or poor execution, see [18]), and, although modularity is a desirable feature from the perspective of combinatorial search in product design (cf. [1], [26]), there often are many (unforeseen) technological relationships to take into account when designing products.

New product development and design (NPD) literature is gradually steering clear from the linear model and incorporating more evolutionary elements [21]. The NPD literature recognizes that product development and design processes (i) may be specific to market, firm, and technological particularities [28], (ii) need to take into account the level of market uncertainty [2], (iii) require product (line) redefinition whenever information on customer needs becomes available [4], and (iv) require recurring testing of market viability [10]. The product design literature is thus developing in the direction of evolutionary innovation economics. In evolutionary economics, firms are engaged in a perpetual process of developing new products to improve the competitive position vis-a-vis competitors by, among others, increasing the attractiveness of products for customers. A pivotal assumption, though, is that firms experience market uncertainty (is there demand for what is being developed?) and technological uncertainty (is the product being developed technically feasible?). Generally, focusing on resolving one type of uncertainty before the other is not commendable. One may end up with a working product for which there is ultimately no demand, or deep insight on what is desired but which is technologically impossible to produce. Moreover, firms suffer bounded rationality in their product design and position decisions. Agent-based computer models of technology competition races, often referred to as neo-Schumpeterian models, need an operational definition of a product (and a digital encoding thereof) as well as the preference of customers for that product. In neo-Schumpeterian models, products commonly are encoded as points in a low-dimensional domain of the function of customer preferences, while innovation is commonly modeled as a path-dependent, random movement of firms in that low-dimensional 'product space' (e.g. [13], [23]).

From the appearance, both the new product design and development literature and the evolutionary innovation economic literature face the same conundrum: it is not clear which design policy to follow. In this paper, we use the agent-based model to study the effects of market and technological structures on the new product design process. Applying agent-based models to understand product development is definitely not new (see e.g. [12]), yet we use it to evolutionary program a periodic product design policy itself, which is new. Hereto, we assume that each product design has technical characteristics which provide services (possibly) desired by the users [20], [25]. Firm agents in our model design a product by puzzling together a combination of platforms and components while user agents can be asked to express the desirability of the services provided by (the components contained in) the product. In this paper, a product is perceived as an assembly of subassemblies, which is represented as a hierarchical, direct graph of platforms and end components (without loops/ recursions). This representation is based on the tree model for products studied in earlier work on product competition [5]–[9] and product innovation in [29], [30]. For the latter two papers, the tree representation hinges on the bill-of-material perspective common in operations management (see e.g. [16], [22]).

III. Agent-based model

Here, we introduce a simple periodic agent-based model of a product-market in which X designers seek to design products to be sold to a set of users $\mathcal{A}$. Unlike in the classical marketing models, designer agents do not know the user preferences, nor the extent of segmentation, let alone the sizes of the various niches. Moreover, designers do not know the structural properties of the technology underlying the product being designed, so experiment with extensions and decide on the next action to take on-the-go. In our model, each designer follows its policy π which specifies for each of the T periods which action to take (notably, whether to gather market information or alter its design). Each simulation run consists of having X designers engage in Y rounds of Z design contests. From one to the next design contest, poorly performing agents imitate design policies of agents with superior performance. As such, the agent-based model 'evolutionary programs' superior design policies producing technologically feasible and market viable designs. Table I contains the notation used in this paper.

Symbol	Default	Description
N	4	Number of platform levels
M_i	$4i$	Number of platforms per level $i = 1, \ldots, N$
$\mathcal{C}$		Complete set of all end components
$L = \lvert\mathcal{C}\rvert$	$(N+1)4$	Number of end components
d		Experimental product design
$\mathcal{S}$		Complete set of services
$K = \lvert\mathcal{S}\rvert$	8	Number of services
$\mathcal{A}_j$		Set of users in niche j
J		Number of market niches
$\mathcal{M}$		Market impression
$\mathcal{C}(S)$		Set of end components required to provide service S
$\mathcal{S}_a$, $\mathcal{S}_i$		Set of services preferred by user a, by users in niche i
$\mathcal{S}(d)$		Set of services provided by design d
$D(d)$		Desirability of design d
U	3	Number connections of platform to lower-level platforms / end components
V	2	Number of components required per services
W	2	Number of services desired per (user in a) niche
T	50	Number of periods
X	25	Number of designers
Y	50	Number of rounds
Z	20	Number of contests
B_z		Technological structure for contest $1 \leq z \leq Z$

Table I: Notation used in this paper

A. Components, technical relationships, technological structure, and product design process

The focus in this paper is on the design process constructing a product from given 'technological pieces'. Whether or not a technological relationship between technological pieces is feasible is specified by the 'technological structure'. We assume that the focal designer does not know this technological structure but has all pieces at its disposal and experimentally puzzles together a product design. We operationally define a *product design* as a combination of components organized in an hierarchical, directed graph, whereby high-level components can contain lower-level components. Both for ease of exposition and operational generation of structures, we discern two types of components, namely (i) 'end components' which cannot contain other components, and (ii) 'platform' components which can contain both lower-level platforms and end components. For the remainder of this paper, we refer to 'components' and 'platforms'. The generic technological structure, as illustrated in Fig. 1, is defined by the number of platform levels N, the numbers $M_1, \ldots, M_N$ of platforms per level, the number L of components, and the U relationships that each platform has with random lower-level components/ platforms ('has place for'). In this paper, the U relationships of each platform at level $1 \leq m \leq M$ are with components/ platforms at the next level $m+1$, i.e. such as how P_{2M_2} is placed on platform P_{12}, and not such as C_1 is connected to P_{12} in Fig. 1.

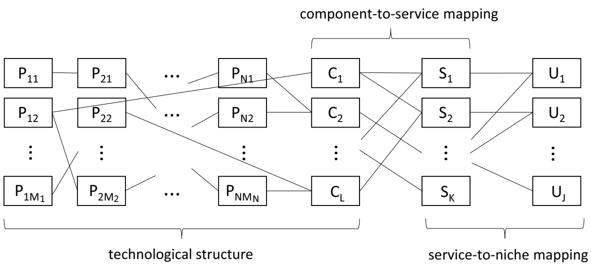

Figure 1: Technological structure consisting of platforms and components, the mapping of components to services, and the mapping of services to user niches.

Designers may take one of two design actions. Firstly, 'top-down' design by starting at a random platform with missing 'children', i.e. with at least one free 'slot' for components or platforms and adding that missing child from the set of components and platforms. Secondly, 'bottom-up' design by starting at a random 'orphan' component or platform with a missing 'parent' (due to which this component or platform is not yet functional) and adding that missing parent. If there are no orphans yet, the market impression $\mathcal{M}$ (which is explained later) is used to determine which component C^* is started with for bottom-up design actions.

In the model in this paper, we implement designers' technological uncertainty in the following ways. Firstly, designers do not know the technological structure and rather than having an advanced algorithm to explore the technological structure, they experimentally puzzle platforms and components together one by one. Secondly, whenever designers pick a (top-level) platform, they do not know which components can ultimately be hosted, and, as such, cannot optimize their choice based on market desirability of the services provided by these components. So, agents suffer 'technological uncertainty' in that they do not know feasible extensions beforehand and do not foresee the technological consequences nor the market viability of choices for technological extensions. However, we assume that designers do know which combination of end components ultimately provides particular services, which is also used in deciding from which end component to conduct bottom-up research (explained in detail later).

B. Services, user preferences, market structure, and market research process

Each product design d provides a particular set of *services* $\mathcal{S}(d)$ which is a subset of a fixed set of services $\mathcal{S}$ possible. Design d provides service $S \in \mathcal{S}$ if d contains the set $\mathcal{C}(S) \subseteq \mathcal{C}$ of components. Which components are required for particular services is specified by the component-to-service mapping (see Fig. 1). In the model for this paper, each service requires one unique component, notably C_i is assigned to service S_i, and then $V-1$ general ones randomly drawn from $C_{K+1}, \ldots, C_L$. As such, the requirement is that $L > K$, i.e. the number of components exceeds the number of services.

Each user $a \in A$ desires W services $\mathcal{S}_a$. The preference of user a for design d is determined by the extent to which the user's service preferences $\mathcal{S}_a$ are met by the set of services $\mathcal{S}(d)$ provided. Users with the same service preferences form 'niches', so all users $a \in \mathcal{A}_i$ in niche i have the same service preferences $\mathcal{S}_i$. The market structure is operationally defined by the number of niches J, the sizes of the niches $|\mathcal{A}_1|, \ldots, |\mathcal{A}_J|$, as well as the number W of services for each niche and the overlap/ distribution of these services over the various niches. The number of users $\mathcal{A}_i$ in niche i is drawn from a power law distribution (with exponent 0.7) such that there are a relatively a few big niches and relatively many small niches. In this paper, each niche is associated with one unique service requirement contained in $\mathcal{S}$ and $W-1$ random ones from the service universe $\mathcal{S}$ (possibly in common with other niches). As such, there is a certain degree of disparity between the preferences of users in different niches. Operationally, this is implemented by having the i-th niche uniquely requiring service S_i and $W-1$ randomly drawn services from $S_{J+1}, \ldots, S_K$. As such, the requirement is that $K > J$, i.e. the number of services exceeds the number of niches.

We implement designers' market uncertainty operationally by initializing that they have no prior information on the number and sizes of niches, nor the service preferences of users in each of these niches. Market research is taken to be 'exploration' of the market to obtain information on (the occurrence rate of certain) user preferences. Operationally, a designer draws a random user a not previously interviewed from the population $\mathcal{A}$ to obtain the set of desired services $\mathcal{S}_a$. The designer then updates its market impression $\mathcal{M}$ defined as:

$$\mathcal{M} := \{(\mathcal{S}_i, p_i) \mid i = 1, \ldots, I\}, \tag{1}$$

which is a list of tuples $(\mathcal{S}_i, p_i)$ for the I readily discovered niches (i.e. unique sets of services S_i), which is so far encounter p_i times.

The *estimated* desirability $\tilde{D}$ of design d is defined as the number of interviewed users for which the services provided by d would be sufficient, i.e.

$$\tilde{D}(d, \mathcal{M}) := \sum_{(\mathcal{S}', p) \in \mathcal{M}} p \, 1_{\mathcal{S}' \subseteq \mathcal{S}(d)}. \tag{2}$$

The real total desirability $D(d)$ is

$$D(d) := \sum_{i=1,\ldots,J} |\mathcal{A}_i| \, 1_{\mathcal{S}_i \subseteq \mathcal{S}(d)}. \tag{3}$$

As mentioned above, in bottom-up design, the market impression $\mathcal{M}$ is used to find the most desirable components not yet incorporated in the design. To this end, the components required for the services desired by users in niches readily discovered receive the weight of the frequency with which (users in) these niches have readily been encountered. As such, we define the *estimated* desirability $\tilde{D}$ of a *component* $C \in \mathcal{C}$ as

$$\tilde{D}(C, \mathcal{M}) := \sum_{(\mathcal{S}', p) \in \mathcal{M}} p \, 1_{C \in \mathcal{C}(\mathcal{S}')}, \tag{4}$$

with $\mathcal{C}(\mathcal{S}')$ the set of components required for the services $S \in \mathcal{S}'$, i.e.

$$\mathcal{C}(\mathcal{S}') := \{C \in \mathcal{C}(S) \mid S \in \mathcal{S}'\}. \tag{5}$$

Upon commencing bottom-up design without existing orphans (see Sec. III-A), the designer takes the yet unexplored component C^* with the highest desirability, i.e.

$$C^* := \arg\max_{C \in \mathcal{C}'} D(C, \mathcal{M}), \tag{6}$$

in which $\mathcal{C}'$ is the set of components not yet incorporated in the design.

Note that, in this simple version of the model, the designer does not have the option to target specific niches and -to this end- optimize the total estimated desirability of a single niche. Here, the designer rather simply targets the entire market and seeks to optimize the (estimated) desirability for the whole market.

C. Evolutionary programming of design policy

Here, we assume that a designer follows a periodic policy π. Each period t, the policy π prescribes the action π_t for the designer to take, where π_t is either market research, a bottom-up design action, or a top-down design action. From the perspective of a sequential design problem, the current decisions are contingent upon future market impressions and future design decisions. As, however, the designer at time t does, evidently, not know the market impression $\mathcal{M}_{t'}$ nor the impression of the technological structure at any $t' > t$ in the future, the current design decision need not be optimal, and an optimal policy cannot be derived.

We use an agent-based model to evolutionary program the 'optimal' design policy. We start with a population $\mathbf{X}$ of designers, each designer $x \in \mathbf{X}$ has its particular policy π_x. This policy is a series of actions encoded as a string of 0's (top-down), 1's (bottom-up), and/or 2's (market research), which is executed repeatedly if the planning horizon is longer than the length of the string. Initially, these policies are uniform randomly drawn. The designers are now involved in Y rounds each consisting of Z design contests. During the series of Z *design contests*, the designers follow the *same* design policy, but upon engaging in a new *round* of contests, they (may) *change* their policy. Each design contest $1 \leq z \leq Z$ consists of T periods in which the designers follow their policies to develop a product design for a particular technological structure B_z. So, for each technological structure B, each designer x comes up with a design d_{Bx} in the last period, which then has total desirability $D(d_{Bx})$.

However, rather than studying only the outcome of a single contest, we use an evolutionary program to find a top design policy after Y rounds of Z contests. Hereby, after each contest, the ρX designers with the lowest total desirability $\sum_{z=1,\ldots,Z} D(d_{B_z x})$ '*imperfectly imitate*' the design policy of a random design of the $(1-\rho)X$ designers with the highest total desirability of their designs. After examination of simulation results, we picked $\rho = 0.8$ to prevent premature convergence (high ρ) and poor selective power (low ρ). In the imitation step, the copying designer experimentally changes the duration of one of the design actions (here: adds or removes one action of the same type). After Y competition rounds, the policy yielding the highest *total* desirability D over all contests is taken to be the winning design policy π^*.

IV. Simulation results

Despite the generality of the model, and the technological structure in particular, we limit the simulation study to two experimental variables, namely one associated with technological uncertainty, and one associated with market uncertainty. Firstly, the number of connections U of a parent platform with lower-level platforms/ end components. Secondly, the number of market niches J. For clarity of exposition, we study 50 cases for the 2×2 contingency table for the number of connections $U = 1, 3$ and the number of market niches $J = 1, 3$. Hereby, we have $|\mathcal{X}| = 25$ designers engage in $Y = 50$ rounds of $Z = 20$ contests (i.e. 20 different technology structures). We thus study the effects of these two experimental variables on the evolutionary emerging ('winning') design policy π^*.

Looking at the extensive simulation results, we see that, in general, if the number of niches J is low, any emerging policy starts with a short market research period. The obtained market impression $\mathcal{M}$ is subsequently used to pick the most desirable component C^* to design into the product through extensive bottom-up design activities. Given that services, here, require, V number of components, the designer may repeatedly determine a new, most desirable component to commence new bottom-up design activities. For $U = 1$, each platform hosts only one lower-level platform or component, and both top-down and bottom-up design will make components feasible in the number of steps required to cross all levels (i.e. $N+1$, with N the number of level). If, in addition to $U = 1$, the number of niches $J = 1$, bottom-up search is efficient in ensuring that the components required for the services for this one niche

	$U = 1$	$U = 3$
$J = 1$	Short market research, followed by bottom-up and then top-down design (21111111111111111111111111111111000000000000000000)	Short market research, followed by almost exclusively bottom-up design (21111111111111111111111111111111111111100000000000)
$J = 3$	Long market research, followed by bottom-up and then top-down design (22222222222221111111111111111111111111111100000000)	Only few cases yield successful policies. However, those feasible have no market research, only top-down design (00000000000000000000000000000000000000000000000000)

Table II: Winning design policy π^* for different settings for the number of market niches J and the number of platforms/ end components U hosted by each platform. A typical example of an emerging design policy is given in brackets, with 2 = market research, 1 bottom-up design, 0 top-down design.

become functional (i.e. are in a contiguous string to a top-level platform). For $U = 3$, each platform hosts three lower-level platforms or end components. Top-down design generally fans out and incorporates platforms and components ultimately not required to provide the services for the niches targeted. Clearly, if there are multiple niches and all are targeted (e.g. in case $J = 3$), many components are ultimately needed, if not for one then for another niche, and top-down design is then much less inefficient. Indeed, we see that in the $U = 3, J = 3$ case, a 'top-down-only' approach is followed by simply not 'wasting' periods on market research and developing a product incorporating as many functional end components as possible through top-down design.

Upon increasing the number of components per layer from $M_1 = 4, M_2 = 8, M_3 = 12, M_4 = 16$ to $M_1 = 6, M_2 = 12, M_3 = 18, M_4 = 24$ without changing the number of components required per service, the number of uninteresting strands to fan out into during top-down design increase. Consequently, this design policy Simulation results corroborate this: emerging design policies are (almost) devoid of 0's (remaining 0's, if any, are to be considered noise rather than distinct phases). In the $U = 3, J = 3$ case, there are no successful policies emerging anymore.

Focusing on this 'top-down-only' policy, we see that first results indicate that this policy is evolutionary unstable for increasing complexity in the market or technology structure. If the number of components per service changes from $V = 2$ to $V = 3$ (many components required), there are no successful policies emerging for a short time horizon. Something different holds if we change the number of services wanted by customers from $W = 2$ to $W = 3$ (many services required). In this case, the time period is too short for the 'top-down-only' policy to develop all W services randomly (and particularly for the bigger niche(s)) and, consequently, no successful policies emerge. If either $W = 2$ (number of services desired by customers) changes to $W = 1$ or $V = 2$ (number of components per service) changes to $V = 1$ (or both W and V change to 1), two different policies emerge as successful: both the top-down-only and a policy starting with a market research, a brief period of bottom-up research, followed by a long period of top-down research.

V. Conclusion

Already from our limited study, we conclude that, in a context with market and technological uncertainties, there probably is no specific 'one-size-fits-all' product design policy. In the evolutionary program featuring repeated contests, we see that designers whom emerge as winners have design policies tailored to the specific segmentation of the market and the density of technological relationships among technologies. While certain rationales apply to the design policies, extreme cases of the market or technological structure may call for radical design policies, e.g. 'just top-down'. We expect that adding more experimental variables and possibly more (refined) market and technological design activities will further enhance our insights in the rationales for design policies.

As there is no real-world counterpart for repeated design contests, our insight that designers need to tailor their design policies stipulates that an online design policy is needed which adapts to (indications of) structural features of technology and market when they become available. Consequently, any mathematical solver maximizing the expected returns by matching technical attributes of products with user requirements may underestimate the implications of technological uncertainties encountered during design. Conversely, investing in establishing technical feasibility prior to having clear indications of market viability may be inefficient.

That all said, grosso modo, a design policy framework generally starts with market research followed by bottom-up design steps. Although this effectively and quickly yields feasible designs for sparse technological structures (say, $U = 1$), the bottom-up design is followed by top-down design steps for more densely connected technology structures (say, $U = 3$). As a critical note, turning to top-down design steps may be said to be caused by the fact that (the current implementation of) bottom-up design creates various loose strands of connected components/ platforms only linked once another top-down or bottom-up design activities discover a platform adjacent to both. Bottom-up design may be more efficient if it *recursively* enables components/ platforms *breadth-first*. Top-down design, in turn, is efficient to create contiguous connections from top-level platforms to bottom-level components for sparse technological structures ($U = 1$). For densely connected technological structures, top-down research is highly likely to fan out in strings of connections not necessarily leading to components required for services targeted. Moreover, these components may still not be functional due to missing parent platforms. Clearly, this becomes less likely if there are many niches (say, $J = 3$), hence explaining the extreme 'top-down only' policy.

Conclusively, with the current setup for the technological and market structure, the diversity of the market determines the extent in which market research is conducted, which should be followed by directed bottom-up design. By analyzing detailed logs of the design activity output, we conclude that bottom-up and top-down design activities and even temporal mixes thereof are generally inadequate design policies. The bottom-up research heuristic should go beyond seeking to establish a contiguous connection of a component to one of the top-level platforms to then commence at the component level again once succeeded. This bottom-up design should be breadth-first and recursive whenever the technological structure is densely connected. In contrast, top-down design may be made more efficient by depth-first rather than breadth-first. For definitive conclusions on this, though, future work should also incorporate depth-first and breadth-first design options.

Moreover, a shortcoming of the current operationalization of the design process (and the criteria therein) is that there are no criteria on the internal structure of a design's component graph if there are alternatives. In the current implementation, it may, for instance, happen that there may be multiple top-level platforms incorporated in the same design to offer different components and thereby services to users, while these components could well be hosted on one and the same top-level platform. In reality, a single, integrated top-level platform is preferred over multiple ones. Finally, an important design activity is to redesign an existing hierarchical graph, and this should be included in future versions of the model.

Acknowledgment

The four co-authors gratefully acknowledge funding from the Deutscher Akademischer Austauschdienst (DAAD), project grant 57219050. Part of the work was done by the two German authors receiving EU H2020 funding, project grant 731726, based on the idea of a hierarchical tree structure conceived while funded by the Dutch Science Foundation NWO, grant 458-03-112, and extended while funded by the German Science Foundation DFG, grant PY 70/8-1.

References

[1] Baldwin, C.Y., Clark, K.B. (2000). Design Rules: The Power of Modularity. MIT Press.
[2] Bhattacharya, S., Krishnan, V., Mahajan, V. (1998). Managing New Product Definitions in Highly Dynamic Environments. *Management Science*, 44(11).
[3] Blond, L., Schiølin, K.H. (2017). Lost in Translation? Getting to Grips With Multistable Technology in an Apparently Stable World, In: Hasse, C., Tafdrup, O., Aagaard, J., Friis, J.K.B. (eds.), Postphenomenological Methodologies New Ways in Mediating Techno-Human Relationships. Rowman & Littlefield International, Lanham.
[4] Chen, S.L., Jiao, R.J., and Tseng, M.M. (2009). Evolutionary Product Line Design Balancing Customer Needs. *CIRP Annals - Manufacturing Technology*, 58(1), 123-126.
[5] Chen, S.-H., Chie, B.-T. (2004). Agent-Based Economic Modeling of the Evolution of Technology: The Relevance of Functional Modularity and Genetic Programming. *International Journal of Modern Physics B*, 18, 23762386.
[6] Chen, S.-H, Chie, B.-T. (2005). A Functional Modularity Approach to Agent-Based Modeling of the Evolution of Technology. In: Namatame, A., Kaizouji, T., Aruka, Y. (eds.), The Complex Networks of Economic Interactions: Essays in Agent-Based Economics and Econophysics, LNMES 567, Springer, 165178.
[7] Chen, S.-H, Chie, B.-T. (2007). Modularity, product innovation, and consumer satisfaction: An agent-based approach. In: Yin, H., Tino, P., Corchado, E., Byrne, W., Yao, X. (eds.), Intelligent Data Engineering and Automated Learning, LNCS 4881, Springer, 10531062.
[8] Chen, S.-H., Chie, B.-T. (2013). Non-Price Competition in a Modular Economy: An Agent-Based Computational Model. *Economia Politica: Journal of Analytical and Institutional Economics*, XXX(3), 149-175.
[9] Chen, S.-H., Chie, B.-T. (2014). Competition in a New Industrial Economy: Toward an Agent-Based Economic Model of Modularity. *Administrative Sciences*, 4(3), 192-218.
[10] Erat, S., Kavadias, S. (2008). Sequential testing of product designs: Implications for learning. *Management Science*, 54(5), 956-968.
[11] Fujita, K. (2002). Product variety optimization under modular architecture. *Computer-Aided Design*, 34(12), 953-965.
[12] Garcia, R. (2005). Uses of Agent-Based Modeling in Innovation/New Product Development Research. *Journal of Product Innovation Management*, 22(5), 380-398.
[13] Gilbert, N., Pyka, A., Ahrweiler, P. (2001). Innovation Networks: A Simulation Approach, *Journal of Artificial Societies and Social Simulation*, 4(3).
[14] Green, P.E., Krieger, A.M. (1989). Recent Contributions to Optimal Product Positioning and Buyer Segmentation. *European Journal of Operational Research*, 41, 127-141.
[15] Green, P.E., Krieger, A.M. (1991). Product Design Strategies for Target-Market Positioning. *Journal of Product Innovation Management*, 8, 189-202.
[16] Hegge, H.M.H., Wortmann, J.C. (1991). Generic Bill-of-Material: A New Product Model. *International Journal of Production Economics*, 23, 117-128.
[17] Kaul, A., and Rao, V.R. (1995). Research for Product Positioning and Design Decisions: An Integrative Review. *International Journal of Research in Marketing*, 12, 293-320.
[18] Van Kleef, E., Van Trijp, H.C.M., Luning, P. (2005). Consumer research in the early stages of new product development: a critical review of methods and techniques. *Food Quality and Preference*, 16(3), 181-201.
[19] Krishnan, V., Ulrich, K.T. (2001). Product Development Decisions: A Review of the Literature. Management Science, 47(1), 1-21.
[20] Lancaster, K.J. (1966). A New Approach to Consumer Theory. *The Journal of Political Economy*, 74(2), 132-157.
[21] Loch, C.H., Kavadias, S. (2007). Managing New Product Development: An Evolutionary Framework. In: C.H. Loch and S. Kavadias, Handbook of Research in New Product Development Management, Elsevier.
[22] Mather, H. (1986). Design, Bills of Material, and Forecasting - The Inseparable Threesome. *BPICS Control*.
[23] Nelson, R.R., Winter, S.G. (1982), An Evolutionary Theory of Economic Change, Harvard University Press.
[24] Piedras, H., Yacout, S., Savard, G. (2006). Concurrent Optimization of Customer Requirements and the Design of a New Product. *International Journal of Production Research*, 44(20), 4401-4416.
[25] Saviotti, P.P., Metcalfe, J.S. (1984). A theoretical approach to the construction of technological output indicators. *Research Policy*, 13, 141-151.
[26] Simon, H.A. (1965). The Architecture of Complexity. *General systems*, 10, 6376.
[27] Ulrich, K.T., Eppinger, S.D. (2008). Product Design and Development. New York: MacGraw-Hill.
[28] Unger, D., Eppinger, S. (2011). Improving product development process design: a method for managing information flows, risks, and iterations. *Journal of Engineering Design*, 22(10), 689-699.
[29] Vermeulen, B., Pyka, A. (2014). Technological Progress And Effects Of (Supra)Regional Innovation And Production Collaboration. An Agent-Based Model Simulation Study.. *Proceedings of IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr)*.
[30] Vermeulen, B., Pyka, A. (2014). The Effects of Supraregional Innovation and Production Collaboration on Technology Development in a Multiregional World: A Spatial Agent-Based Model Study. In: J. Was, G. Sirakoulis, and S. Bandini (eds.), *Cellular Automata*, Lecture Notes in Computer Science, 8751, 698-707