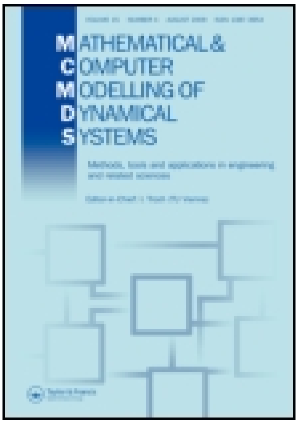


Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/nmcm20

In-process agent simulation for early stages of hospital planning

Gabriel Wurzer [a]

[a] Digital Architecture and Planning , Vienna University of Technology , Vienna , Austria

Published online: 17 Jan 2013.

Mathematical and Computer Modelling of Dynamical Systems, 2013
Vol. 19, No. 4, 331–343, http://dx.doi.org/10.1080/13873954.2012.761638

In-process agent simulation for early stages of hospital planning

Gabriel Wurzer*

Digital Architecture and Planning, Vienna University of Technology, Vienna, Austria

(*Received 16 June 2012; final version received 19 December 2012*)

In the early stages of hospital planning, work processes are typically modelled in a static manner by using flow charts or business process modelling notation as means. Diagrams of this kind are easily simulated; however, employed process engines lack possibilities for dealing with dynamic aspects of the process which depend on the building layout (e.g. elevators, behaviour of automatic delivery carts). If one could give planners the opportunity to employ dynamic entities without having to change their usual workflow, one of the benefits would be that they are not being forced to resort to naive assumptions (e.g. 15 seconds per floor) that are still commonplace in today's planning practice. As another contribution, we show how agent-based simulation can be used as an analysis tool by using the state of the process simulation to generate arrivals.

Keywords: hospital planning; functional design; process simulation; agent-based simulation; hybrid

1. Introduction

Hospitals, like airports and some types of industrial facilities (e.g. oil platforms), are process-driven buildings: Their design depends foremost on the planned work processes that enable them to operate day and night, 365 days a year. Therefore, the process model of such a building constrains the architectural design, which must evolve in close cooperation between process planners and architects.

Because processes are modelled in a highly formalized manner (e.g. as flow charts), one might think that the application of simulation lies at hand from the very start of a building project. However, such *static* process descriptions lack the ability to also include aspects that *depend on the building layout*, such as the transition of persons and material from one space to the other, possibly using dynamic entities such as lifts as they move along. Resorting to naive assumptions (e.g. fixed passage times) might be inadequate (again taking the lift as example) and, furthermore, cumbersome to elaborate. In early planning, there are usually *several variants* of the spatial concept rather than only one for later phases.

Our work therefore focuses on overcoming the mentioned problems by embedding dynamic entities into an otherwise static process model. Broken down into further detail, our contribution consists of:

- A thorough look at 'simulation needs' in the early stages of process-driven building design (see Section 3). Such a survey is surprisingly novel, as the community has

*Email: gabriel.wurzer@tuwien.ac.at

previously targeted hospital simulation problems but not their context within the planning process.
- An extension of static process simulation such that dynamic entities (acting in a spatial context) can be represented. Technically, this is achieved by invoking an agent-based simulation on behalf of the process simulation (see Section 4).

Our work here is purely foundational: We want to augment the now-common working style of planners in a non-intrusive manner, that is, extending rather than reinventing design tools available. The choice of an agent-based simulation *on top of* a process simulation fits exactly this line of reasoning. As a matter of fact, we do not focus on results, but on the concepts needed to obtain them. If process modelling tools were extended in the spirit proposed herein, it is likely that a paradigm shifts away from separated planning of organizational processes and functional/spatial design towards a more integrated planning practice occurs. Apart from reducing the communication overhead, this also translates to faster turnover times. Especially when it comes to architecture competitions, this can be a very significant factor.

2. Related work

Business process simulation (BPS) is based on linking a graph-based model to a discrete event simulation (DES) that simulates its behaviour over time. There exists a variety of software packages implementing BPS (e.g. [1,2]) plus some DES packages that provide a ‘flow chart’ like approach by means of a server/client-based model [3–6].

When it comes to simulation that requires an understanding of the spatial concept (as in the previously mentioned examples of elevators and automated delivery carts), we see that most of the DESs seem to focus on late design phases, that is, phases in which the spatial concept is already fixed and subject to optimization (e.g. via [7]). Especially in hospital planning, this might be a problem, since spatial design is subordinate to the process model, and might thus not be evolved as far as the latter when a simulation is performed. We have, therefore, previously presented a coupled pedestrian/process simulation targeted at early design stages, in which the early concept (also called schema) is taken into account [8]. Our efforts for this paper are approaching the problem from a different side: Our goal is to enable planners using a static modelling approach (i.e. flow charts) to include dynamic entities into their process descriptions, based on a BPS being linked to an agent-based simulation (ABS). We are aware of many approaches being occupied with this specific hybrid mode [9,10]. However, none is focused on the planning context of early-stage hospital design, which is essential when producing an approach that is adapted to working routines now in place.

3. Background

The simulation needs for early-phase hospital design are closely connected to the design process. In the following subsections, we will describe the typical planning tracks and deliverables in early design, before coming to the actual problem areas in which simulation can provide valuable input when being used as a design tool. Because of space constraints, we have omitted a discussion on the influence of different design methods used, and may forward readers interested in this topics to [11,12].

3.1. The early design process

There are two design tracks that are important to early-phase hospital design, *Building Organization* and *Functional Planning*. The brief overview of these tracks that follows is important, since they form the context of work routines within which an 'early-stage' approach needs to operate.

3.1.1. Building organization

Building Organization (see upper part in Figure 1) is occupied with the planning of the organization from the side of business administration, that is, definition of the *organizational structures* (departments, sub-departments), *work processes* and *responsibilities* within these. In essence, the planning work proceeds top-down: starting with a very coarse outline of business activities required for operation (Figure 1(a)), a basic formulation of processes can be derived by introducing temporal and causal order (Figure 1(b)). The notation of these processes depends on preferences of the project team, two usual options are flow charts or process graphs according to the recently standardized business process modelling notation (BPMN).

As the project progresses, some activities might need to be further detailed in order to be fully defined. This can be done by using sub-processes, which establish a hierarchy of activities within activities (Figure 1(c)). Furthermore, when detailing a process, responsibilities for each activity are also assigned to different collaborating departments within the organization (see Figure 1(d)). The finished product and goal of the building organization is thus a description of the whole operation of the building from a business side (also called 'process model' of the organization). The process model acts as input and constraint for the functional planning track.

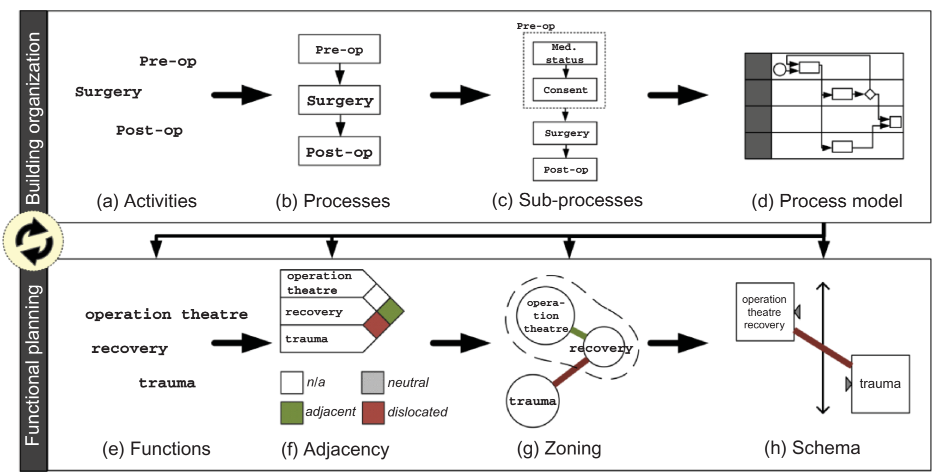


Figure 1. Early-stage planning tracks. (upper part) Building Organization: (a) activities formed into (b) processes and (c) sub-processes. Furthermore, assignment of process responsibilities to different departments leads to (d) process model, which acts as input and constraint for (lower part) Functional Planning: (e) Functions are (f) related via an adjacency matrix, (g) grouped to form (h) spaces within the architectural schema. Circulation is additionally inscribed using arrows.

3.1.2. Functional planning

Functional Planning (see lower part of Figure 1) starts with a definition of building functions (i.e. capabilities of a building, see Figure 1(e)), based on the intended vision (laid out e.g. in the tender document, project description, etc.) and process model of the organization. These functions are then correlated in an adjacency matrix [13] by the degree of collaboration, ranging from 'adjacent' for closely collaborating areas to 'dislocated' for areas that do not cooperate or must be separated, for example, because of hygienic considerations (Figure 1(f)). Adjacent functions are then grouped into *spaces* (refer to Figure 1(g)): in the example given, 'operation theatre' and 'recovery' are put into one common space (signified by a dashed border), while 'trauma' stays isolated and gets its own space. The so-found spaces are then arranged in a preliminary floor plan called 'schema' (see Figure 1(h)), with each space being represented by a rectangle.

In this context, the rectangular form of every space is not to be taken literally, since it merely gives the proportion, approximate size and location in relation to other spaces. The concrete form for each space is beyond the work done in early-stage design – it occurs later, in a phase called 'Form Finding'. Apart from the spaces, the schema also contains arrows that give the preliminary circulation system (e.g. corridors) of the building. Aside from the graphical notation, the schema is typically also given as spreadsheet form ('Space Allocation Plan'), which listing spaces (often grouped by function), cardinality (e.g. 2x) and usable area per space (e.g. 25 m^2), commonly regulated by guidelines and planning handbooks such as [14]. Depending on project structure, the Space Allocation Plan may be produced during Functional Planning or be given before the actual planning work starts, as an input (e.g. when extending a building).

3.1.3. Co-evolution through design decisions

It is noteworthy that the activities of Building Organization and Functional Planning are not sequential but inherently *parallel*: As process model and the spatial concept are detailed and evolve side-by-side, the planning team has to ensure consistency of both models. Furthermore, the spatial concept might fork of a variety of alternative designs, which must then later be reduced or merged by design decisions (i.e. documented argumentation within the planning team leading to a set of choices, see [15,16]). For process-based buildings in the early planning stages, these design decisions are typically based on:

- *Urban context*. It is the relationship between the planned building as a whole (i.e. arrangement of spaces and circulation) with its surrounding environment and the existing infrastructure [17]. For example, traffic patterns resulting from local public transport and motorized individual traffic have to be taken as a constraint. Visibility of landmarks has to be preserved by (and likely used in) the proposed design.
- *Adjacency*. Short paths between collaborating units (defined by the adjacency matrix), considering the adjacency matrix [13], process model and expected volume of building users requiring service. Vice versa, a separation of spaces for reasons of privacy (e.g. secure areas vs. public spaces, inpatient vs. outpatient areas) and for sustaining building operation (typically by service corridors, allowing for repairs and delivery 'behind the scenes').
- *Separation of traffic*. Different routing according to type of traffic [18], for example, separation of staff from visitors and patients, low-priority from high-priority traffic (e.g. emergencies), building users with appointment from the ones without, soiled from clean material, and so on.

- *Location, size and proportion*. Placement of spaces is not isolated from considerations of the building as a whole; for example, certain areas of work favour natural lighting (e.g. patient rooms, energy considerations for the whole building), while others can do without. Proportion and size of individual spaces determine the opportunity of future adaptations (e.g. change in equipment), while at the same time being subject to optimization (minimal area needed per function, compactness).
- *Orientation and way-finding*. Depending on intended user spectrum, orientation can play a vital role for the whole building project. The transition of building users from one space to the other must be considered both in terms of the process as well as existing previous knowledge about the building layout. Spaces serving processes used by temporary building users must be easy to reach (i.e. no signage required) and memorize (e.g. using a main corridor connecting all departments). A clear readability of space can also help in fire safety and evacuation planning, conducted in later phases [19].
- *Extensibility and adaptability*. Both extensibility and accessibility of a building is given by the configuration of spaces and circulation [14]. The first one deals with openness to the outer environment, and the second with (usually multi-functional) hub spaces that serve as distribution points for pedestrian traffic, often located at prominent positions within the building. The ability to adapt the spatial concept to future requirements of the process also requires an evaluation from a multi-functional view (e.g. interdisciplinary use of a space, shared workspaces, etc.).
- *Adequacy of planned concept*. The adequacy of both spatial concept and process model is an overall judgement of the building's design under consideration of the planning task. Argumentation focuses on whether the design satisfies the vision and financial context stated by the client. In the planning team, the discussion is centred on the types of functions present and sizes of their respective spaces as well as structure of the processes and needed resources.

3.2. Early-stage simulation needs for hospitals

Given the mentioned design decisions in early planning phases of process-based buildings, simulation can contribute tools for assessing a variety of aspects which can then be weighted according to the planning objectives (i.e. multi-objective analysis). Statements produced in this manner are necessarily qualitative, since spatial concept and process model are in a preliminary stage.

3.2.1. Visibility, accessibility and way-finding

The analysis of these parameters may be done statically (for the whole building, its arrangement of spaces and circulation network) or dynamically (by simulating individual processes). In the first case, reachability analysis of the circulation network can be conducted for both interior and exterior spaces by using the methods provided by Space Syntax [20,21], which can also compute the visibility from each point in the building (e.g. for hiding areas for supply and disposal). View shed computations, usually found in geographical information systems (GIS), can be used for the same purpose. Way-finding, on the other hand, requires a dynamic simulation of individuals following their processes (e.g. using ABM). In this connection, algorithms from pedestrian dynamics may be used to simulate the physical movement under the influence of congestion (e.g. via [22]).

3.2.2. Space placement and dimensioning

Previously defined adjacency relations can be verified by simulating the planned processes by means of BPS, ABS or system dynamics (SD). The volumes of traffic between the spaces, distances travelled over the circulation and simulated times taken must correlate with the relationships given in the adjacency matrix. Furthermore, the dimensioning of spaces can be checked by considering the volume of persons present in each time step: In the simplest case, the occupation is related directly to the presence of persons in a space (e.g. in entrance areas). Moreover, presence in a space may relate to waiting for a shared function (e.g. examination), which can be modelled as server with a specified number of resources (e.g. two doctors) and one or more queues. By correlating the observed size of queues with the space requirements for waiting areas (distinguished e.g. for sitting and lying patients), it is possible to attain a hint at minimal areas required. Norms and regulations further contribute to these space requirements, which could be checked using approaches from automated building code checking [23], albeit in a simplified form.

A further opportunity for comparing the placement of spaces is that of building physics simulation: Some workplaces might require daylight; others must be protected from it. Preliminary environmental simulation (light, shadowing, wind), can also hint at energy demands which are elaborated in later phases.

3.2.3. Movement, circulation and traffic

Different options of route choice can be simulated by either assigning way-points between subsequent activities of the process explicitly (e.g. delivery of goods in zigzag shape, one floor at a time) or by interpreting the circulative network as graph on which shortest paths are computed. As a matter of fact, spatial arrangements can be judged by the time it takes to move across the circulation (which is also depending on the processes in place). A separation of traffic can further be achieved by attributing the circulation with allowed types of traffic (e.g. patients, visitors, staff), and taking these into account during either automatic or manual route planning. A further attribution of the circulation arrows has to be performed for distinguishing between horizontal traffic (taking place in the same level) and vertical circulation (lifts and stairs), among which movement models and speeds might differ.

3.2.4. Usage

Functions give the purpose (or intent) of spaces, processes model their planned usage over time. By coupling activities of the process to the underlying functions, a static check for unused function or activities that have no reference to a function (i.e. the underlying spatial concept) can be made [8]. Furthermore, temporal usage of functions obtained via process simulation can be used to compare the prominence of the spaces containing them and help think of possibilities for multi-functional use: Areas that are used only part-time (e.g. lunch room) may be conveniently used for other functions (e.g. meeting room) during the rest of the day.

4. An early-stage hybrid simulation

Our hybrid approach extends process simulation through agent-based simulation in two fundamentally distinct ways:

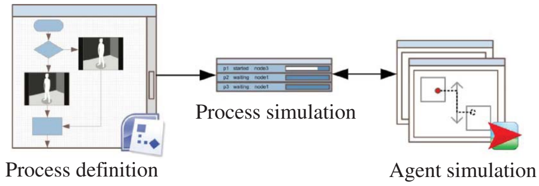


Figure 2. Overview: Processes are fed into a BPS, which communicates bi-directionally with a set of ABS performing the actual 'dynamic' work.

- *Intrinsically* by adding dynamic behaviour to nodes inside a process (see Section 4.1).
- *Extrinsically* by invoking external simulations that perform analysis, using the current state of the process simulation to generate arrivals (see Section 4.2).

Figure 2 gives a general outline of the model that we are employing: A process definition is fed into a BPS, which communicates bi-directionally with an ABS. BPS and ABS differ in their time bases (discrete event versus simulation in seconds); therefore, some synchronization is required. In simplest approach (which we have undertaken), the BPS' scheduler was customized to progress in seconds instead of advancing to the next future event. We are aware that this is a very odd intervention – after all, discrete event simulation is about having a future event list that controls progression of time. One can, however, constrain the 'progression in seconds' to cases where ABS are active, in which case the BPS are required to wait anyway.

4.1. *Adding dynamic behaviour to a static process*

In order to inject dynamic behaviour into a static process, we introduce a new type of activity coined as 'agent node' (see Figure 3(a)): This is essentially a proxy for an agent simulation that is to be executed on behalf of the BPS. There are different types of agent nodes, with each one standing for one dynamic aspect that is missing from the static process.

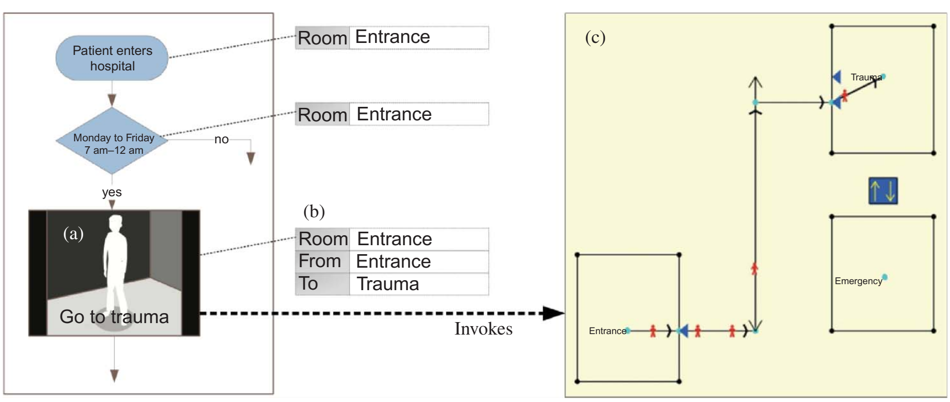


Figure 3. A process in which (a) an agent node was embedded. All nodes are (b) annotated with a room stamp. Agent nodes furthermore hold additional parameters that are (c) passed to the underlying ABS.

The example we have implemented is an ABS of pedestrian movement through the architectural schema (see Figure 3(c)); but generally, there will be several types of these dynamic models running in parallel, each having their own type of agent node and respective ABS running in the background. The agent node itself holds parameters specific to its model (e.g. 'from' and 'to', see Fig 3(b)). Process simulation is then performed in the following manner:

(1) Upon entering an agent node, the current process is passivated.
(2) A message is broadcasted to the ABS connected with that agent node type, which the process ID and parameters mentioned before. The ABS creates an agent having the given process ID and initializes it with the parameters given (in our case: computes route along the circulation, from the source to the target space).
(3) After all processes have been executed for this time step, the process simulation sends a message to the ABS to start simulating this time step and waits until this is done. This effectively hands control over to the ABS.
(4) After receiving the command to start simulating, the ABS advances by 1 second. It then sends a message back to the process simulation that it has finished, also containing the IDs for all agents that have reached their goal (i.e. finished simulating).
(5) After receiving the finishing messages of all ABS, the BPS reactivates processes with the given IDs and goes to the next simulation round.

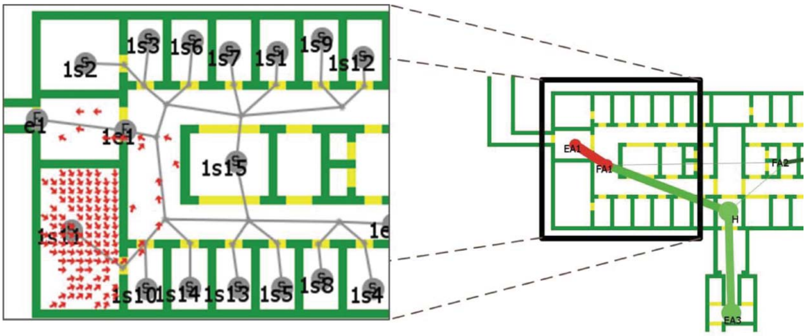
Some further logic may be also embedded into the BPS to announce process creation and termination, for example, creating and removing agents at the same time as in the BPS (*eager creation*), and to use agent nodes in order to assign tasks. In this manner, information can be stored within each agent, which can use it accumulatively over the course of several calls to the same model.

4.2. Using processes as inputs for dynamic analysis

The proposed simulation approach has so far only used ABS intrinsically, that is, as delegate that executes dynamically *within the process*. By extending the approach to pass the set of active processes *as arrivals* for an ABS that runs externally, one can use it as analysis tool covering the aspects mentioned under Section 3.2. As illustrative example of such a tool, we have implemented a throughput visualization (see Figure 4(b)) that computes flows through functional areas by utilizing a pedestrian simulation algorithm ([24], see Figure 4(a)).

As pre-step and requirement for this extension, the nodes of the process have to be attributed with room stamps (see Figure 3(b)). The ABM must be able to derive physical locations within the design from these – for example, by using lookup table or through interrogation of the design itself. In full detail, the model consists of the following steps:

(1) The BPS samples all processes at time t. For every process, the room stamp attached to the executing node is recorded. Counting the total number of times that it appears then yields the occupancy per room stamp, which is then broadcast as message to all ABS. The BPS again can continue simulating, since the analysis is running completely detached (i.e. *extrinsically*).
(2) Upon receiving a simulation request, the ABS create agents in the given rooms according to occupancy. The agents are arranged randomly and until the capacity



(a) (b)

Figure 4. Example of ABS used for external analysis of a process. (a) Computation of pedestrian movement, based on arrivals given by the state of the BPS. (b) Analysis of flow depicted as coloured lines of varying thickness, connecting the different functional areas of the hospital.

of the containing space is reached (in which case an error can be reported saying that this room is under-dimensioned).

(3) With all agents being in place, the simulation starts and computes its respective aspect. In the example given in Figure 4(a), agents perform an egress simulation to the nearest reachable exit. The simulation records the flow through each area they pass, and visualizes the throughput as 'flow lines' with thickness according to pedestrian volume (Figure 4(b)).

In practice, t will correspond to 'interesting' points in the operational schedule of the hospital (e.g. early morning, morning, mid-day, early afternoon, night/weekends). Differences between the results produced at times t_i are especially interesting for looking at the building under different usage scenario which vary over the day. For example, the variation in throughput over a set of arrival times can be depicted as colour of the flow lines – red for large variations and green for minor variations.

4.2.1. Implementation details

We have extended a commonly used process modelling platform (Microsoft Visio^{TM}) for which a multitude of BPS exists (e.g. ProModel^{TM} Process Simulator, Simul8^{TM} and Arena^{TM} Integration). However, because we needed to customize the scheduler, we coded our own BPS in Java^{TM}, based on the open-source graph-based programming code provided by [25]. As ABS platform, we have employed NetLogo [26], which is also available under open-source license terms.

In brief, the system we have implemented works as follows: we first start all ABS. Then, we let Visio^{TM} output the process model as file with which the BPS is invoked. From then on, the BPS communicates quasi-bi-directionally with all ABS in the following fashion (refer to Figure 2): ABS act as servers, listening on a certain port for incoming messages. The BPS is a client, issuing a request at regular intervals (e.g. 0.25 seconds). To buffer messages, BPS and ABS each have two message queues (incoming and outgoing). One communication round is as follows: in each request, (1) the BPS removes one message from its outgoing queue and (2) sends it to all ABS. If there is no message to transmit, the BPS issues a '*nop*' (no operation, nothing to be done) message instead. The ABS (3) receives the message by the BPS and puts it into its incoming queue lest it is *nop*. It then (4) responds to the request by removing one message from its outgoing queue or sending *nop*. The BPS (5) receives the message and puts it into its incoming queue, if it is not *nop*. At both sides, the incoming and outgoing queues are manipulated by the programs, the sending occurs automatically.

The exchanged messages themselves correspond to the scenarii we are supporting. For intrinsical simulation, we can *create agent with a certain ID in given room* (e.g. 'process 12 entrance') or *simulate one second* ('step 1'). The extrinsical simulation understands *create occupancy in room* (e.g. 'occupancy entrance 9'), *simulate until completion* ('start'). The ABS knows only how to announce that it has finished simulating ('end').

The actual implementations used for the ABS are based upon the previous work [27–29] and fall beyond the scope of this article: In essence, what we want to show is the feasibility of the 'in-process' approach, and not reflect on already published material. What is common to both ABS is that they interact with the preliminary design (e.g. exported from AutoCAD™ by plotting at a resolution of 0.5 m per pixel, then colour-coding and importing into cell space).

5. Including the toolkit into early-stage design

Under Section 3.1, we have presented the early design process of hospitals from the viewpoint of the two main professions involved (Building Organization and Functional Planning). In this section, we intend to elaborate how exactly the presented concepts fit into the co-evolutionary planning work and can help take design decisions (see Sections 3.1.3 and 3.2). As will be shown, intrinsical simulation is for the simulation of individual activities, whereas extrinsical simulation is intended for analysis.

In both cases, processes and preliminary spatial concept need to be available: They act as a model for the overall simulation. The input (i.e. patient arrivals) is to be generated either by use of an assumed distribution or, preferably, by reference to actual patient volumes (e.g. from a similar clinic). These data are extractable from an Hospital Information System (HIS), although quality may vary; the usual case is that it is known when a patient arrives (timestamp) in a certain functional unit (e.g. radiology). From this information, the arrivals can be reconstructed to a certain degree. However, and as a word of warning, examination durations are not adequately recorded, as waiting times in between consecutive stages are not explicitly mentioned. The best approach is therefore to measure service times at the functional unit directly.

5.1. Applying intrinsical simulation

Intrinsic simulation approaches the planning problem from the side of Building Organization, using BPS as primary and ABS as secondary/subordinate simulation. This mode is synchronous as seen from the BPS side, since the ABS executes until the agent that stands for the process instance is released. The proposed Agent Nodes are delegates that execute *dynamic activities* that depend on and influence the spatial concept of the building. Referring to Section 3.2, these tasks range from way-finding processes of building users under consideration of visibility and accessibility (Sections 3.2.1 and 3.2.3), possibly also influencing the dynamically changing space layout and dimensioning (Section 3.2.2) to questions of resource allocation and queuing (Section 3.2.4). This means that ABS used in this context to serve a single activity and purpose, with no synchronization in between multiple ABS running in parallel. In fact, these models might not even be made visible to the user without request – the primary tool is the BPS.

5.2. Applying extrinsical simulation

Extrinsical simulation stops execution of processes at time t and uses simulation to gather performance indicators over the spatial concept under the current occupancy at that instance. The questions presented under Section 3.1.3 are the context in which this analysis happens. However, we have so far left out *how* to actually measure and analyze these factors, given the setting that the process and spatial model used in early planning is by definition preliminary, not to be assessed quantitatively but rather qualitatively. As an example, adjacency specification between each two functional areas can be specified as ‘adjacent’, ‘neutral’, ‘dislocated’ and ‘not applicable’ (see Figure 1). The analysis of a spatial configuration under the influence of the simulated process would first try to map the distance between each pair of spaces along the circulation to these linguistic categories, and then weight that result according to simulated traffic (i.e. ‘high traffic’, ‘medium traffic’, ‘low traffic’). Both points can be conducted efficiently by using fuzzy architectural spatial analysis [30], which represents a linguistic term by a membership function and outputs a continuous likeliness in [0,1]. This output would then be either presented directly or be interpreted automatically using, for example, a rule-based system that outlines the competition requirements. A good choice for such a rule-based evaluation that results in yet another (derived) membership function would be the domain of Fuzzy Inference Systems (FIS), for example, the Mamdani-type or Sugeno-type FIS.

6. Outcome, summary and conclusions

We have presented a simulation approach by which dynamic entities can be embedded into an otherwise static process. Our efforts are targeted at early stages of hospital planning, where simulation has the potential to become a design tool for qualitative decisions among a multitude of design variants. We have presented two different modes for such ‘in-process’ agent-based simulation: (1) *intrinsical simulation*: introducing nodes that add dynamic behaviour and (2) *extrinsical simulation*: passing all active process instances to and agent-based simulation that performs a (spatial) analysis for the sake of decision-making in the design process, for example, for comparing different spatial configurations. The latter simulation needs in early design have been argued for in an own part (see Section 3.2), which is also a contribution in health care design, in which we see the true impact of our work: The possibility to shift simulation from late stages of design (where possibilities of change are limited) to early stages (where design decisions are of fundamental significance) continues to be our main goal in research.

Acknowledgements

We would like to acknowledge the support given by Niki Popper, Martin Bruckner and Felix Breitenecker who have been of vital importance since the early phases of this paper. Furthermore, we wish to thank Simon Peyton-Jones and Jan Michl for their contribution to scientific presentation and discourse.

References

[1] B. Rücker, *Building an open source business process simulation tool with JBoss jBPM*, Master Thesis, Stuttgart University of Applied Science, 2008.

[2] S. Dodds, *Designing improved healthcare processes using discrete event simulation*, Br. J. Healthc. Comput. Inf. Manag. 5 (2005), pp. 14–16.

[3] D. Sadowski and V. Bapat, *The arena product family: Enterprise modeling solutions*, Proceedings of the 2001 Winter Simulation Conference, Arlington, VA, 9–12 December, ACM Press, New York, 2001, pp. 159–166.
[4] W.B. Nordgren, *Taylor enterprise dynamics*, Proceedings of the 2001 Winter Simulation Conference, Arlington, VA, 9–12 December, ACM Press, New York, 2001, pp. 269–271.
[5] W.B. Nordgren, *Flexible simulation (flexsim) software: Flexsim simulation environment*, Proceedings of the 2003 Winter Simulation Conference, New Orleans, LA, 7–10 December, ACM Press, New York, 2003, pp. 197–200.
[6] K.H. Concannon, K.I. Hunter, and J.M. Tremble, *Simul8-planner simulation-based planning and scheduling*, Proceedings of the 2003 Winter Simulation Conference, New Orleans, LA, 7–10 December, ACM Press, New York, 2003, pp. 1488–1493.
[7] M. Laguna and R. Martí, *The OptQuest callable library*, Oper. Res/Comput. Sci. Interfaces Ser. 18 (2003), pp. 193–218.
[8] G. Wurzer, *Schematic systems – constraining functions through processes (and vice versa)*, Int. J. Archit. Comput. 2 (2010), pp. 201–217.
[9] M. Remondino, *Agent based process simulation and metaphors based approach*, 4th Workshop on Agent-Based Simulation, Montpellier, France, 28–30 April, SCS European Publishing House, Erlangen, 2003, pp. 93–97.
[10] A. Borshchev and A. Filippov, *From system dynamics and discrete event to practical agent based modeling: Reasons*, *techniques*, *tools*, Proceedings of the 22nd International Conference of the System Dynamics Society, 2003. Available at http://www.systemdynamics.org/conferences/2004/SDS_2004/PAPERS/381BORSH.pdf (Accessed 13 December 2012).
[11] J.C. Jones, *Design Methods*, Wiley & Sons, London, 1992.
[12] B. Lawson, *How Designers Think, the Design Process Demystified*, Architectural Press, Oxford, 2005.
[13] E. White, *Space Adjacency Analysis: Diagramming Information for Architectural Design*, Architectural Media, Tucson, 1986.
[14] E. Neufert, *Architect's Data*, Blackwell Science, Malden, 2000.
[15] W. Kunz and H. Rittel, *Issues as elements of information systems*, Working Paper 131 (1970), Center for Urban and Regional Development, University of California, Berkley.
[16] H. Rittel and M. Webber, *Dilemmas in a general theory of planning*, Policy Sci. 4 (1984), pp. 155–169.
[17] E. White, *Site Analysis: Diagramming Information for Architectural Design*, Architectural Media, Tucson, 1983.
[18] P. Lohfert, *Methodik der Krankenhausplanung*, Lohfert & Lohfert AS, Copenhagen, 2005.
[19] C. Illera, M. Fink, K. Kath, N. Waldau, A. Rosic, and G. Wurzer, NO_PANIC. *Escape and panic in buildings*, Proceedings of the 4th International Conference on Pedestrian and Evacuation Dynamics, Wuppertal, Germany, 27–29 February, Springer, Berlin, 2008, pp. 733–742.
[20] B. Hillier and J. Hanson, *The Social Logic of Space*, Cambridge University Press, Cambridge, 1984.
[21] B. Hillier, *Space Is the Machine: A Configurational Theory of Architecture*, Cambridge University Press, Cambridge, 1996.
[22] D. Helbing and P. Molnár, *Social force model for pedestrian dynamics*, Phys. Rev. E. 5 (1995), pp. 4282–4286.
[23] C. Eastman, J.-M. Lee, Y.-S. Jeong, and J.-K. Lee, *Automatic rule-based checking of building designs*, Autom. Constr. 18 (2009), pp. 1011–1033.
[24] V. Blue and J. Adler, *Cellular automata model of emergent collaborative bidirectional pedestrian dynamics*, Proceedings of Artificial Life VII, Portland, MD, 2000.
[25] T. Baeyens, *JBPM: Graph Oriented Programming*, 2005. Available at http://docs.jboss.org/jbpm/v3/userguide (Accessed 13 December 2012).
[26] U. Wilensky, *Netlogo*, Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, 1999. Available at http://ccl.northwestern.edu/netlogo (Accessed 13 December 2012).
[27] G. Wurzer, M. Ausserer, H. Hinneberg, C. Illera, and A. Rosic, *Sensitivity visualization of circulation under congestion and blockage*, Proceedings of the 5th International Conference on Pedestrian and Evacuation Dynamics, Gaithersburg, MD, 8–10 March, Springer, Berlin, 2010, pp. 899–902.

[28] G. Wurzer, *Early-stage egress simulation for process-driven buildings*, Proceedings of the 6th International Conference on Pedestrian and Evacuation Dynamics, 2012, to appear in print.
[29] G. Wurzer, W. Lorenz, and N. Popov, *Meeting simulation needs of early-stage design through agent-based simulation*, Proceedings of the 30th International Conference on Education and Research in Computer Aided Architectural Design in Europe, Prague, Czech Republic, 12–14 September, eCAADe and ČVUT, Prague, 2012, pp. 613–620. ISBN 978-9-4912070-2-0.
[30] B.C. Arabacioglu, *Using fuzzy inference system for architectural space analysis*, Appl. Soft. Comput. 10 (3) (2010), pp. 926–937.