Heriot-Watt University

Empiric design evaluation in urban planning

Aschwanden, Gideon D. P. A.; Haegler, Simon; Bosche, Frederic Nicolas; Van Gool, Luc;
Schmitt, Gerhard

Link to publication in Heriot-Watt Research Gateway

Empiric Design Evaluation for Urban Planning

Augmenting generative 3D City Models with Behavior-Based Agents

Gideon D. P. A. Aschwanden,
Chair for Information Architecture, ETH Zurich, Switzerland
aschwanden@arch.ethz.ch

Simon Haegler,
Computer Vision Laboratory, ETH Zurich, Switzerland
shaegler@vision.ee.ethz.ch
Procedural Inc., Zurich, Switzerland

Frédéric Bosché,
Computer Vision Laboratory, ETH Zurich, Switzerland
bosche@vision.ee.ethz.ch

Luc Van Gool,
Computer Vision Laboratory, ETH Zurich, Switzerland
vangool@vision.ee.ethz.ch

Gerhard Schmitt,
Chair for Information Architecture, ETH Zurich, Switzerland
schmitt@arch.ethz.ch

ABSTRACT: We propose a system to simulate, analyze and visualize occupant behavior in urban environments, combining parametric modeling and agent-based simulation. A procedurally generated 3D city model, containing semantic information about the functions and behaviors of buildings, gets automatically populated with artificial agents representing pedestrians, cars and public transport vehicles interacting with the built environment and each other. The system identifies empiric correlations amongst: functions of buildings and other urban elements, population density, utilization and capacity of the public transport network, and congestion effect on the street network. Practical applications include the assessments of a) bottlenecks, b) public transit efficiency, c) amenities accessibility, d) level of service of public transport and the traffic network, as well as e) the stress and exhaustion of pedestrians. All these aspects ultimately relate to the quality of life within the given urban areas.

KEYWORDS: Artificial intelligence, Procedural Modeling, agent based simulation, Space Syntax, urban planning, design evaluation, occupant movement.

1. Introduction

The majority of the world population already lives and works in cities (UN-Habitat, 2009). This influx of new residence puts a lot of pressure on the existing infrastructure, and on the planning of new and upgrade of existing areas of the city. Cities, like Laos, can't keep up building the necessary infrastructure, why the quality of life remains generally insufficient. This is partially confirmed by the level of stress experienced among citizens when compared to rural inhabitants. Even doe inhabitants of cities are consuming less energy then rural dwellers (Cai, 2008), cities still consume too much energy, produce too much waste and emit too much CO2 to constitute a sustainable way of living. As a result, we are facing the unprecedented challenge of simultaneously improving the livability and sustainability of cities.

Sustainability and quality of life are both complex matters that depend on numerous other, sometimes conflicting, aspects. In the last century, urban planning patterns placed emphasis on path and network optimization for cars and made drastic changes to the structure of the city. The effect of these changes not limited to the individual traffic or other street users, it also changes the allocation of amenities, land price etc. Adjusting one aspect of the city has an influence on different other equilibriums within the city. It now appears clearly that optimizing the urban organization from the point of view of pedestrians positively impacts both the sustainability of the urban environments and the quality of life of their citizens. A shift in the mindset has thus been taking place with the human perspective shifting in the focus of attention. In the context of pedestrians, we present a robust and efficient method for simulating and visualizing the performance of different urban environment alternatives. This method uniquely combines the techniques of commercial crowd simulation applications with the iterative strength of procedural city modeling techniques, which enable: a) assessments of the impact of a given built environment on pedestrians, and b) efficient iterative analysis of different built environments. This enables planners to efficiently search more subtle ways to adjust the urban fabric and gain a human perspective. It must be noted that the automation aspect of our

method also offers an added value for the entertainment industry while delivering high quality imagery output through standard production pipelines and a decreasing workload on the procedurally generated urban layouts that are simulated as a potentially realistic urban environment with associated virtual occupants. Traditionally, costs and time needed to produce populated digital urban sets are enormous for movie, game and interactive VR projects.

The rest of the paper is organized as follows. After reviewing background work in the field of city modeling and urban simulation (Section 2.), Section 3. will present the system that we propose for the simulation of pedestrians within a city environment, and assess the impact of the built environment on pedestrians, and vice versa. Finally, the performance of the proposed system is analyzed through 3 examples (Section 8.).

2. Related works

This project combines works from different fields and has therefore to reference different streams of research. We organized this part accordingly in 2.1 Crowd Simulation, 2.2 Urban Planning and 2.3 ,The rest of the paper is organized as follows. After reviewing background work in the field of city modeling and urban simulation (Section 2.), Section 3. will present the system that we propose for the simulation of pedestrians within a city environment, and assess the impact of the built environment on pedestrians, and vice versa. Finally, the performance of the proposed system is analyzed through 3 examples (Section 8.).

2.1. Crowd Simulation Overview

Models for crowd behavior have been an active research field since the late 19th century (e.g. see LeBon, 1895). Today's computer simulation models have a relatively young history. Most relevant approaches have been realized within the last 20 years and are specialized to different fields. More specific is Reynold's (1987) flocking method, which uses particle systems and represents one of the most common approaches for simulating group movements.

For multi-agent and large crowd simulation, several techniques have been proposed to animate or simulate large group of automated agents and crowds. These methods can be classified in five groups (Pelchano, 2008):
- **Potential-based methods:** Pedestrian agents are modeled as particles with potentials and forces (Helbing, 2008)
- **Boid-like methods**: These approaches were introduced by Reynolds, creating simple rules for computing the velocities (Reynolds, 1999)
- **Geometric method:** The aim of these approaches is to compute collision free paths. They either integrate the velocity space or use optimization methods (van de Berg, 2009, Guy, 2009)
- **Field based methods**: These algorithms generate fields for agents to follow, or generate navigation fields for different agents based on continuum theories of flows or fluid modes (Treuille, 2006, Narain, 2009)
- **Least effort crowds**: These algorithms compute the paths of crowd agents using Zipf's (1949) principle of Least Effort (Zipf, 1949). Recently these have been combined with collision avoidance algorithms and emergency behaviors for a large number of agents (Patil, 2009).

In detail, Brooks (1991) provides a comprehensive foundation on which many of the recent agent models and theories are based. He describes many failing artificial intelligence approaches to set-up intelligent agents. Musse and Thalmann (2001) introduced a more flexible model with hierarchical behavior. Physics and body effects had been described by Helbing et al. (2000) to simulate escape behavior and panics effectively. In other fields, like robotics (Molnar and Starke 2001), safety science (Still 2000) and sociology (Jager et al. 2001), similar approaches have created simulations involving groups of individual intelligent units. For a more comprehensive description of agent based pedestrian movement, we refer to Magnenat-Thalmann et al., (2004). Furthermore, Hillier and Hanson (1984) introduced the idea that a city and spaces in general can be divided into components to analyze them as a network of choices and be represented as maps and graphs. Penn and Turner (2002) then described their method to use urban agents within their space syntax system. Parallel systems evolved in computer graphics to populate urban environments in real-time (Penn und Turner, 2001; Tecchia and Chrysanthou, 2000).

2.2. Urban Planning

Current planning methods are based on the experience of the planner and end when the architect finished the building. This is insufficient for the future needs, changes, and increased pressure on the performance of urban configurations. There is a demand for simulations, incorporating a wider range of scales, interdependencies – from building codes to the level of the entire city, in order to evaluate and predict the impacts of planning efforts and changes over time. Single layer optimizations, such as emergency evacuation or traffic simulations, exist in many forms and are widely commercially available. These tools incorporate a very limited number of parameters and either

focus on the building performance, neglecting the interdependencies of buildings, or work in the scale of a metropolitan region, using a grid with a scale compiling whole areas of a city.

Recently, there have been approaches to combine behavioral and structural modeling (Vanegas et al. 2009). These approaches treat the behavioral state as static input. To find the equilibrium between them, an automatic iteration of this loop is still far from being an integrated, interactive system. A closer integration of function, behavior, and state reduces calculation time and allows a practical integrating.

2.3. Pedestrian movement in Urban Planning

There is an effort in transport planning to include pedestrian movements and to move from car to pedestrian - oriented transport planning. However, this development is currently still hampered by a lack of empiric simulation methods. Current methods for pedestrian movement analysis are meant for us in early design stages and thus treat pedestrians more like a statistical input: common parameters are land use and modes of transportation. According to the classification of crowd simulation methods (Section 2.1), commercial software for pedestrian flow analysis tend to either use a particle systems with a social force model, neglecting the spontaneous decision processes and that people actually walk and trip, or incorporate simulated movements of crowds, but this requires knowledge of the position of every source and sink, which has to be manually incorporated.

2.4. Functional Urban Models

Wegener (1994) proposes urban models, in which he defines interactions between different entities such as land use, networks, population, house, employment and transportation of goods, and then uses the model to compare existing operational models. Wu and Webster (1998) developed a model integrating multi-criteria evaluation with cellular automata to simulate land use and land use changes. Cellular automata include the special features of traditional urban models and capture the spatial features of the urban fabric. Artificial intelligence (AI) has been adopted to use a knowledge-based methodology so that models can be constructed in a compositional way and to provide the ability to simulate decision-making processes of a single agent or a group of agents (Batty and Xie, 1997; Arend et al., 2001).

Waddell and Ulfarsson broadly define urban simulation as "operational models that attempt to represent dynamic processes and interactions of urban development and transportation" (Waddell and Ulfarsson, 2004). Their introduction gives a good overview of specific techniques (such as cellular automata or multi-agent systems) that have been successfully implemented in a state-of-the-art urban simulation system, UrbanSim (Waddell, 2002). Traditionally, the tools operate on the level of regular grids considerably larger than individual building lots. More recent approaches overcome the rigid grid cell model by taking more refined scales into account, typically starting from parcels to zones (Waddell et al., 2008).

3. Overview of Proposed Approach and Contribution

In Aschwanden et al. (2008), an occupant simulation method was introduced, which is now extended by the present work. The procedural modeling technique, on which our urban model is based, was initially presented by Parish and Mueller (2001) for the modeling of cities. Then, Mueller et al. (2007) developed it for the modeling of buildings, and it has since been practically applied to the context of urban planning by Ulmer et al. (2007) and Halatsch et al. (2008a and 2008b). Generative city models not only allow adjusting and updating the physical representation of the city and its street layout or a single building, but they also incorporate metadata about the use, construction details and capacity of most of the elements constituting the cities. With the knowledge of floor space, function and location (which city, distance to the centre) of every building, we are thus able to predict the amount of traffic (e.g. cars, pedestrians, etc.) every building generates and absorbs over the course of a day. For example, the number of pedestrian agents generated by a given building agent relates to its size, function and location. The combination of methods from artificial intelligence incorporates also behavior of these items. For example, agents are used to represent not only the buildings according to the master plan but also their adaptive functions. Every building agent therefore has some meaning to other agents. Every agent has an individual set of preferences towards specific places and amenities and has individual abilities, like strength, endurance, speed etc., to reach them. On the other hand, each point of interest is adaptive. For example, a parking lot loses its attraction for further car-agents if it's occupied – empty parking lots will gain their attraction again. Primarily every moving agent has a major goal and visits other places along the way. Agents find their way through the city and constantly interact with other agents and the environment. Their experience is color coded in an individual line, analyzing their path towards internal and objective attributes, ranging from empiric, e.g. total time of travel, to personal, e.g. effort to achieve all goals. Each agent then draws an individual path on the ground, color-coded according to a set attributes defined by user, e.g. exhaustion, amount of personal space etc. It is important, when analyzing these pathways, to be cautious about an evaluation,

which comes from a single agent having problems. However, we know that, if hundreds of agents are having the same problem in the same area, the problem is inherited in the design (Surowiecki, 2007). We can measure 'the level of exhaustion' of agents, which then enables us to put public transport stations or resting areas accordingly. To avoid unnecessarily long distance travel to a single goal for an agent, we can propose changes to the traffic guidance system or the street network itself.

3.1. Overview of Proposed Approach

The proposed approach is based on agent based crowd simulation (Massive software package). At the center of the simulations are pedestrian agents that are created in relevant locations in the city and attached with goals (final and intermediary) and behavior models. The generation locations and the goals of the agents are automatically established based on the functions of the buildings constituting the city model. Then, the city model is augmented with a public transport network with moving agents (e.g. buses). These agents are also provided with goals and behavior models. They impact the simulation through their predefined services and their limited service capacity. Finally, the system simulates car traffic in order to take into account its impact on both the pedestrian and the public transportation traffics.

In order to make the intended simulation possible and realistic, a detailed and semantically rich city model is necessary. For instance, the city model should clearly specify building functions (used for defining their capacities to generate and absorb pedestrian agents) and street characteristics (sidewalk size, number of car lines, crossroads characteristics). Further, as the ultimate goal of this work is to support urban planning and thus the assessment of the impact of the characteristics above on the overall traffic, it is important that the city model enables changing these characteristics easily. This motivated the use of generative modeling. In our approach, we use the CityEngine software package to create parametrically a city models. The digital city model is not limited to the 3d representation and includes semantic information of the buildings and the street network.
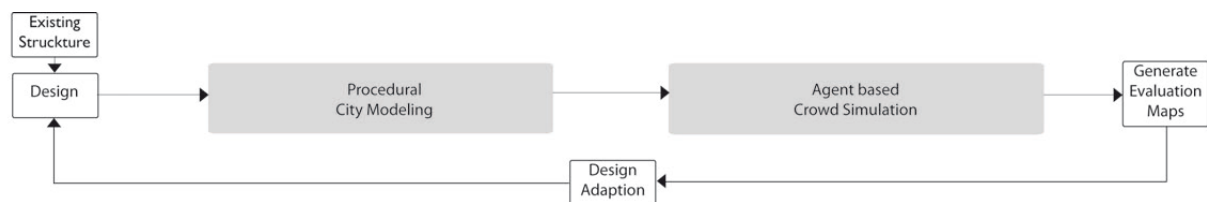

FIG. 1: Workflow for a design iterations (Input – Output based systems).

3.2. Contribution

The contribution of this work is thus multiple. First, the relationships between all urban agents are more accurately modeled. The pedestrian-agents are attributed with dynamic (i.e. adaptive) behavior models with specific preferences for specific final and intermediary goals and individual abilities (such as strength, endurance, speed etc.) to reach them. The goals are themselves given adaptive behavior models that depend on internal and external dynamic parameters. This results in a more realistic modeling of the dynamic interaction of the urban environment and the pedestrian agents.

The second contribution of this work is a method for creating city models, containing physical representations of their constituting elements and detailed networks of functions that can be directly imported and used in the simulation software package. This significantly impacts the integration and the level of automation of the overall simulation process. The city models can be effectively created from existing information such as master plans, street networks, and elevation maps.

The third contribution is a method for automatically defining the locations where agents must be generated from the semantic information contained in the city model imported into the simulation platform. This further impacts the efficiency and robustness of the overall simulation process.

Finally, we provide a means to visually assess the performance of the simulated urban environment, enabling urban planners to identify areas where improvements could be made and to test alternative solutions. The generative parametric city model and the level of automation achieved with the proposed system are critical to enable the planners to effectively and efficiently conduct such an iterative process.

4. Urban Model

We organize the build environment of the city in 2 layers: I) the physically build environment and II) the semantic distribution of functions. Both layers are dynamic but have different time scales, e.g. a building can change its function

several times over its life cycle, the same function of a building is used with different intensities over the course of the day etc. Therefore, we interpret the term 'city' as a complex, distributed, interconnected and rapidly changing system that can be understood as a fabric of space depending on a) people, b) function, c) space, and d) physical environment. Each aspect listed above is typically under the influence of a number of variables and forces. As urban planners, our aim is to try to control both the physical layout and the function contained in the urban system.

In the case of a complex city space, the population may either accept the designers' work or not, but only its subsequent use of the designed urban space will show whether the urban plan was successful. In urban planning there exist many important correlations. These call for a balance, which must be reassessed constantly. For this simulation we are required a) to understand the given situation, b) to identify the necessary semantic metadata and c) to incorporate the insights into the procedural 3D model and the agents' model. The goal is to estimate as accurately as possible the acceptance of the given configuration of a city by the users.

4.1. Density

Urban densities are under constant discussion - the factors by which to classify density are not finally concluded. In addition, the output value does not have a clear unit. Some urban planners find that the number of occupants per square kilometer is a useful figure. Another useful figure may be the number of cars recorded between urban centers, but these two figures have already very little correlation. Therefore, a major achievement would be a metric correlation between the built environment, number of people and traffic. Our contribution is a first empiric explanation of this correlation. We use the existing plots, with those being developed, to predict the number of people likely to be occupying that area. Other calculation methods, like Arnott and Small (1998), use functions of the distance from a virtual centre or a grid overlaid on the city. The 'grid-cell' and 'virtual-centre'-method are rough, contain an imprecision and only produce useful results in large metropolitan areas.
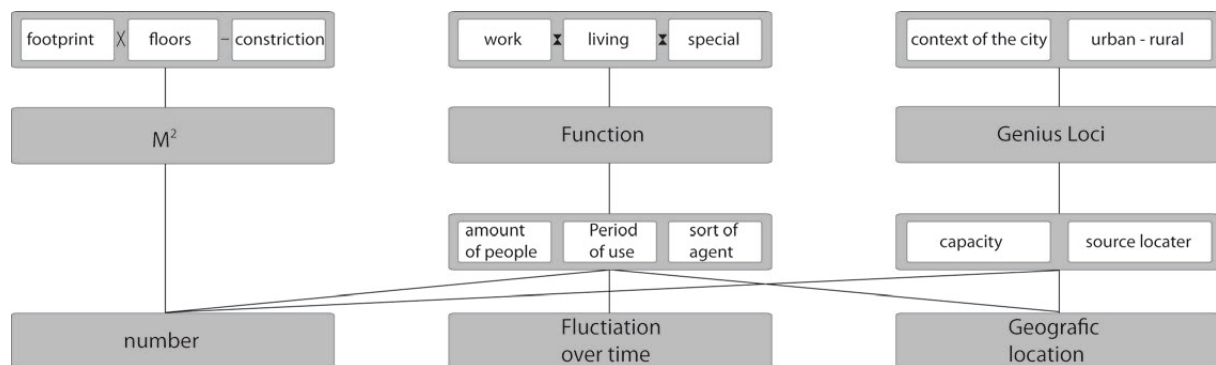

FIG.S 2: Human Locator (Input-Output based calculation method for the prediction of the number of people coming out of a building).

In our approach (Fig. 2), we want to know how many people are coming out of the building and are walking the streets. There are several aspects involved; three of them have a major influence: a) the floor size of the building, b) the function and c) the location of the building. These factors enable us to predict how many people are coming out of a building at a certain moment in time. The statistics is key to comprehensive results, the 'Bundes Amt für Statistik' of Switzerland provided us with the statistical input data of every municipality in Switzerland. The lack of foreign data is limiting our method to Switzerland.

We used the Swiss census data by the Swiss government to define the floor space per person in a building living area as well as the space requirements for offices and workspaces. The overall number of occupants is not decisive enough, only in combination with the fluctuation over time, it tells us how many people are moving in and out of the building. We automated the calculation and specified it according to every commune in Switzerland (Diner, 2006). This was necessary due to the high disparity of the input data. For residential areas we used the floor space of the apartments and divided it by the number of residents to get the average space available to every resident. The combination of net floor space, space available, location and the fluctuation over time allowed us to predict the traffic produced by every building. The available average floor space per resident varies in Swiss cities from 35.5m2 (Zürich) to 55.7m2 (Maisprach). Both numbers are higher than the space provided for office workstations from a minimum 4.46m2 for secretarial workstations to 27.89m2 for a vice-president office. This shows that the traffic generated by office buildings is much higher than the residential buildings in general and is also more volatile over time. We note that other places, such as restaurants, cinemas as well as hospitals, produce even higher numbers in their daily operation time peaks, but it is difficult to predefine them. The limits of such automated calculation are reached in cases where the traffic generated is not a stable function, such as a stadium emptying itself within 30 min twice a week or when the number of buildings is too small to balance out the statistical imprecision.

4.2. Urban Environment

Most recently, research in architecture (and subsequently computer graphics) has produced a number of production systems for architectural models, such as Semi-Thue processes, Chomsky grammars, graph grammars, shape grammars, attributed grammars, L-systems or set grammars (Vanegas et al., 2009). All of these methods expose different application possibilities and levels of efficiency to the user. The shape grammar concept has recently been made more applicable to computer graphics and daily usage (Mueller et al., 2006) and is now commercially available in the software package "CityEngine", which is used in this paper (Procedural Inc http://www.procedural.com). The key tools of the context-sensitive shape grammar as implemented in the CityEngine consist of: shape operations for mass modeling, component splits for transition between mass and facade, split operations for building facades and spatial query operations and spatial query operations.

The workflow to create a virtual urban model for crowd simulation is similar to the workflow used by architects in urban planning. Based on a number of maps (color-coded images, raster data or vector maps) parametric rules are triggered to create street-networks and rough volumetric models of the buildings. These volumes are used to distribute urban functions and building density and to guide the agent navigation as described in the previous section. Based on these rough building volumes, we create the final detailed geometry needed for visualization and the low-polygon geometry used in simulation. The original aspect of our workflow is the use of procedural modeling to automate the creation of street networks and building geometries. A big advantage of this is the adaptability - the urban environment is able to change for each simulation without redrawing from scratch. We automatically export the data set for the simulation and the visualization, which differ radically from one another. They both are stored inside the same procedural scene description and will therefore always be in sync. The following sections describe our workflow in more detail.

4.2.1. Creating the Urban Layout

We use a fictional scenario in the area of Zurich, Switzerland, to describe our workflow. In this example, we use different maps (Fig. 3) to encode aspects of the urban layout (e.g. topography, elevation, obstacles, skyline, land usage). To create the street network, we have two options: (1) generate a generic, procedural (i.e. rule-based) network which follows the attribute maps (e.g. along the color gradients) or (2) import a network from a vector data source (e.g. from openstreetmap.org). For simplicity, we use a generic street network. The algorithm is an extension of the L-system based street network generator described in Parish and Mueller (2001). The building plots can be created in a similar way: (1) by a generic subdivision algorithm or (2) by importing a vector data source. Again, we show generic parcels , also created with the algorithm described by Parish and Mueller (2001). Once the building parcels have been generated, procedural rules can be applied to create the building and the street geometry.
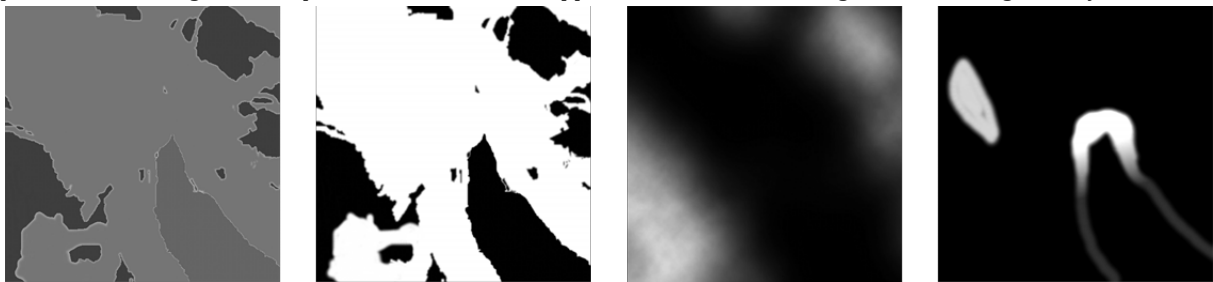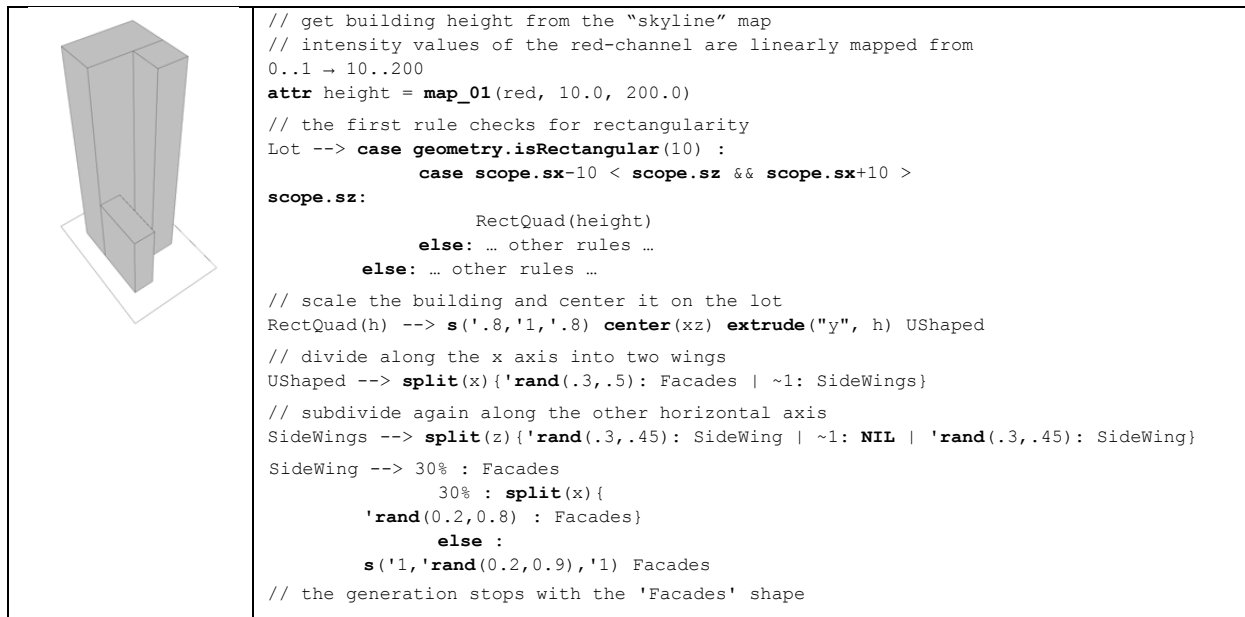

FIG. 3: Colour-coded attribute maps are used to control the generation of street-networks and building geometries. a) topology b)obstacles c) height map d) building heights (skyline).

4.2.2. Procedural Modeling using Shape Grammars

We now show how the shape grammar, as implemented in the CityEngine, is used to model some simple building volumes for the Zurich scene. Each building lot (parcel) is assigned a shape grammar rule set. A rule set consists of production statements in the form (Figure 4):

Predecessor → [case Condition1:] Successors 1 [case Condition2:] Successors 2 [else:] Successor N

```
                          // get building height from the "skyline" map
                          // intensity values of the red-channel are linearly mapped from
                          0..1 → 10..200
                          attr height = map_01(red, 10.0, 200.0)

                          // the first rule checks for rectangularity
                          Lot --> case geometry.isRectangular(10) :
                                      case scope.sx-10 < scope.sz && scope.sx+10 >
                          scope.sz:
                                          RectQuad(height)
                                      else: … other rules …
                                  else: … other rules …
                          // scale the building and center it on the lot
                          RectQuad(h) --> s('.8,'1,'.8) center(xz) extrude("y", h) UShaped

                          // divide along the x axis into two wings
                          UShaped --> split(x){'rand(.3,.5): Facades | ~1: SideWings}

                          // subdivide again along the other horizontal axis
                          SideWings --> split(z){'rand(.3,.45): SideWing | ~1: NIL | 'rand(.3,.45): SideWing}

                          SideWing --> 30% : Facades
                                       30% : split(x){
                                  'rand(0.2,0.8) : Facades}
                                          else :
                                  s('1,'rand(0.2,0.9),'1) Facades
                          // the generation stops with the 'Facades' shape
```

Left: FIG. 4. A simple U-shaped mass model consisting of three volumes (shapes).
Right: Listing 1. This "CGA shape" source code produces simple U-shaped mass models consisting of three shapes. Some parts have been omitted for simplicity.

With this the successors can be composed of several shape operations and query statements. Listing 1 contains a small rule set example and Figure 4 shows a possible result. Figure 5 shows the result of the application of the example rule set example to the complete scene.
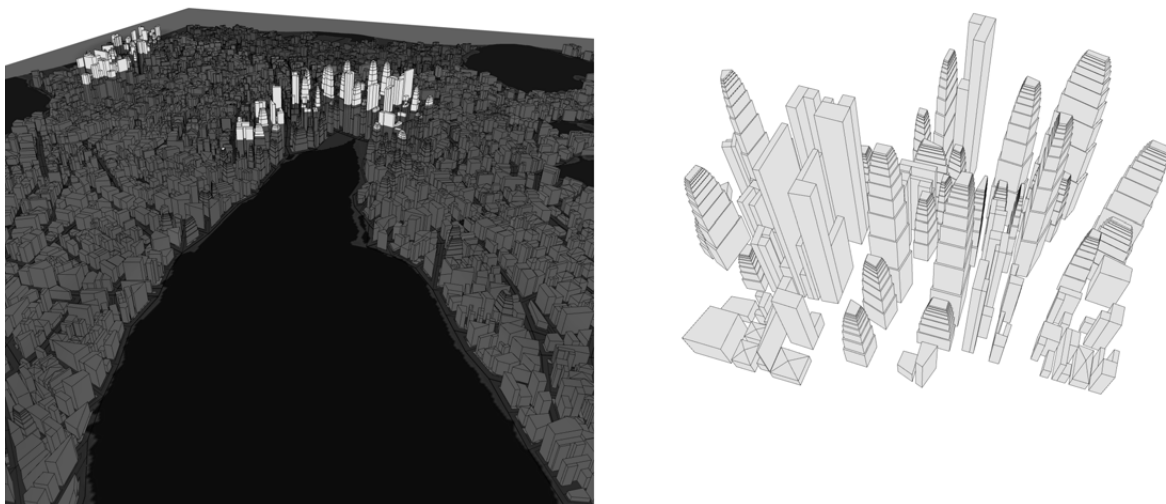

FIG.S 5: On the left, the resulting scene is shown with the same simple set of shape grammar rules applied to all building lots. The figure on the right shows a close-up of the high-rise area in the center.

5.   Generation of the Control Data for Crowd Simulation

As detailed later in Section 6.2, our crowd-simulation-setup needs four types of input from the city model to initiate the simulation:

1. Simplified building and street geometry to visually guide the agents in order to prevent collisions.
2. Locators for initial agent placement and points of interest to implement building functions.
3. Directed colored street lanes to guide cars and buses.
4. Terrain geometry with a texture map where additional color-coded control data is stored. For example, we are using color intensity in the door areas to encode the capacity and function of a certain building.

We describe here how these channels are derived from the procedural urban model. In other words, we need to extend our grammar rule set with an optional set of rules, which trigger the generation of control data for the simulation.
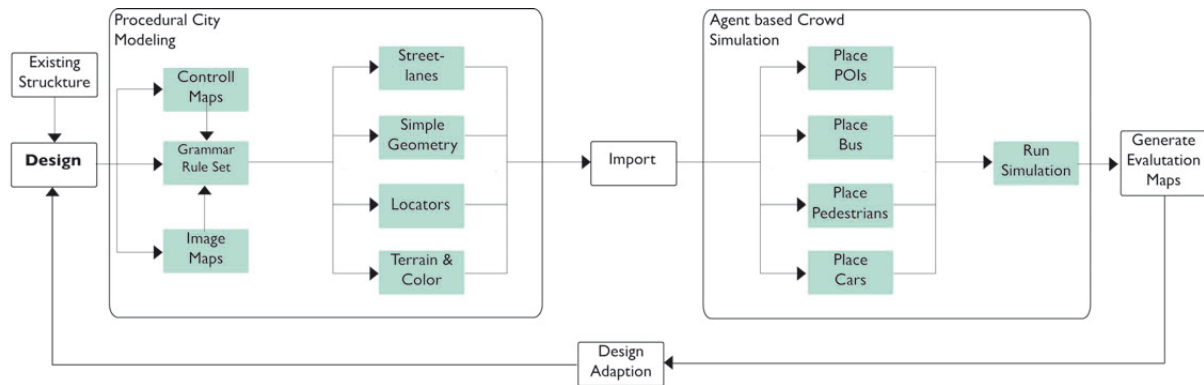

FIG. 6: Illustration of the data-flow between the CityEngine (left) and the crowd simulation tool (right). Note that all the necessary input data to the simulation have been generated automatically from a single grammar rule set.

An export algorithm is developed within the CityEngine which scans the names of the terminal shapes of the grammar generation process (in the U-shaped building example above these were called 'Facades') and triggers the creation of one of the input data types for the crowd simulation based on a set of name patterns. For a typical agent simulation scenario the relevant terminal shapes are usually called 'door', 'window', 'sidewalk', 'street lanes' etc. Figure 6 depicts the data-flow between the CityEngine and the crowd simulation tool. The necessary modifications to an existing grammar rule are:

- Based on the simulation scenario and the simulation tool (in our case "Massive"), choose the appropriate control features (vision, sound, colors, vector fields, etc.) to trigger certain actions of the agent (walk, stand, wait, etc.).
- Identify the parts of the urban environment with which the crowd agents interact (e.g. doors, sidewalks etc)
- Introduce rules, which trigger certain simulation features not necessarily visible when rendering the scene (e.g. a locator on the bottom face of a building entrance). The main idea is to make these rules conditional on a rule attribute to be able to comfortably switch between model generation for visualization and model generation for simulation.
- Also introduce conditional rules to deactivate all geometry not needed in the crowd simulation.

For our final results, the previously mentioned modifications resulted in the following control data for the simulation readable by the agents:

- Spline segments for street lanes and turnings on crossings. We use the underlying street graph connectivity to connect different street lanes on crossings.
- Locators for agent sources: all terminal shapes called 'door' trigger the generation of a locator and a color on the bottom-face of the doors. The color is used to encode the amount of agents emitted from this door.
- Locators for points-of-interest (POI): the doors also trigger a sound source, which is connected to it to attract agents. We distributed 10 different frequencies for the ten different POINTS-OF-INTERESST s (POI0,...,POI9) according to the master plan. If necessary, the distribution of the POINTS-OF-INTERESST s can also be controlled by grammar rules.
- Locators for "background" agent sources: the shape grammar splits the sidewalks into small stripes and assigns agent sources to them according to a certain probability (see Figure 7).
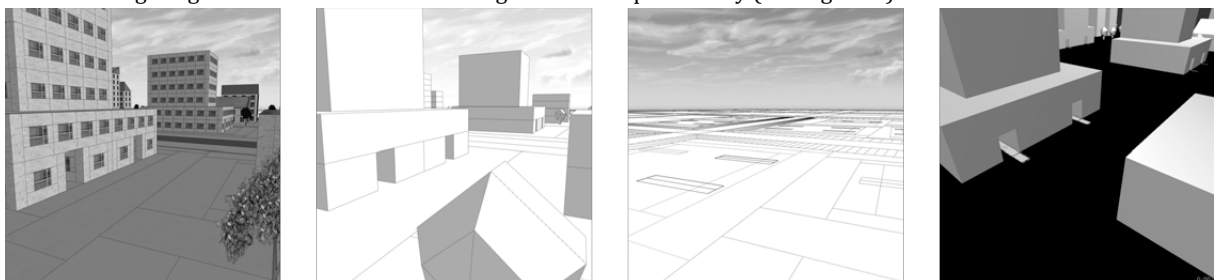

FIG. 7: From left to right: (1) Detailed geometry used for visualization. (2) The rough volumes used for agent vision in the simulation. (3) The blue/red marked areas show the 'door' terminal shapes used to generate agent source locators. (4) Volumes (gray), color-code (red) and locators (yellow) imported into the simulation tool ("Massive").

6. Agents

In order to test the practicality of the modeled semantic 3D city models, the agent-based simulation is conducted with agents representing pedestrians, buildings, cars and mass transit. Every agent represents a measuring device of the city, using its perceptive channels to learn about the environment and the artificial brain to process it and give us information about the path it took. The agents consist of perceptual, behavioral and cognitive components in combination with a physical body to evaluate the practicality of our design and its correlating semantic 3D city model. The agents have perceptive channels and are able to see and hear their environment and read information that has been 'written' to the ground of the 3D environment in a color map or colored lanes. These abilities are processed inside the artificial intelligence for each individual agent at every instance (25 fps). According to their evaluation they emit sound, tag the ground and change their color. This enables agents to interact with their surrounding, their built environment as well as with other agents. They interact individually in a dynamic way (e.g. building-agents lose their attraction for cars and pedestrians temporarily if they reached their capacity). A secondary-level of interaction is needed in some cases like between buses and pedestrians a bus stop indicates whether a pedestrian-agent is waiting.

6.1. Sensory Channels for Perception

We are using multiple sensors as receptors for each agent to understand its environment. Like humans the agents have a limited set of senses through which to experience the environment. The following steps of filtering, selection, rating and simplification have been incorporated into each agent's artificial intelligence to enable it to operate within and interact with their surroundings:

Vision: The vision measures the distance and orientation from a section of the agent (e.g. heag) towards all geometries around it and defines their color. Some agents are using vision for collision detection. The definition of distance (vision.z) in relation to the agent's direction (vision.x) and height (vision.y) in combination with the color (vision.h) of an object or other agent allows the agent to define if this is a threat or not.

Sound: Sound always carries a frequency and is limited to a distance. Each agent who perceives the sound is able to determine the direction (sound.x), distance (sound.d), frequency (sound.f) and emission radius and amplitude (sound.a) of the sound source.

Ground: The ground consists of two parts, geometry and texture map. The geometry is used to determine the global height of each part of the agent (ground.z). The ground color can be read absolutely (ground.r /.g /.b) as well as its change (ground.dr /.dg /.db).

Lane: For streets we use lanes, consisting of a spline with width and color (lane.h). The agent can determine its absolute position on the lane in relation to the centre (lane.x), the deviation of its own angle towards the angle of the spline (lane.ox) and the rate of change (lane.ax).

Every channel has specific implications for the agent. Reading the ground allows the pedestrian agents to understand where they walk to avoid different parts or change their perceptive model, e.g. while crossing the street taking more attention, they pay more attention laterally where cars are likely to appear.

Vision is very calculation intensive but offers numerous opportunities, for instance by putting information as well as on vertical walls for agents to read. Hearing, on the other hand, is very calculation efficient and therefore used as the common interaction channel between all agents.

6.2. Set of Goals / Points of Interest (POI)

Several analytical tools have been presented which examine the correlation between how people move within, and use, the urban environment (Penn and Turner, 2001). In our approach this urban environment is not static. It is a dynamic environment reacting to its surrounding, adapting to changes in the flow of occupants. Each pedestrian-, car-, and mass transit agents has its individual set of preferences in relation to the specific places it wants to visit. These places would be work and home but also the places he visits in between. By compiling such information we reduce buildings to symbols of their functions. We incorporated the allocation of these Point of Interest (POI) in the rules of the procedurally generated city – triggered by the master plan which consists of function layers, such as living, working, industry etc., and creates an additional set of locations for special activities – generating a physical building and a building-agent (Point-of-Interests) (see Section 5). Each building is also defined as an agent that emits a specific sound representing its function, and thus attracting other agents. The capacity of a building-agent is determined by its size, position within the city and function. If the capacity is reached, the building-agent changes its sound emission to be no more attractive until one agent leaves it.

6.3. Decision Process

Pedestrians and car agents are equipped with preferences and a decision process to determine which point-of interest to visit first. With the sound channel the agent is able to define the distance between it and the point-of-interest. The overlay of distance and preferences gives the agent the point-of-interest he wants to visit first. If the point-of-interest is losing his attraction, the decision is recalculated, possibly leading the agent to a different POINTS-OF-INTERESST that he was having as a secondary choice at the start.

7. Specific Agents

7.1. Pedestrians
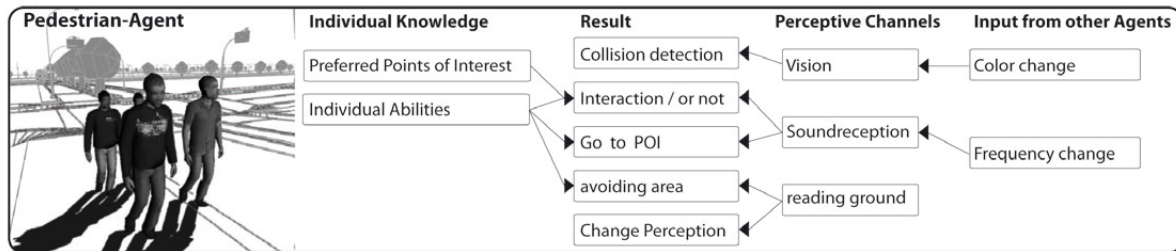
FIG. 8: Input – Output details for Pedestrian Agents

A unique set of personality characteristics and abilities for each agent allows this collection of individual bodies to realistically represent a standard population. Each agent's body and brain constitute a semantic organism enabling the agent to interact with distinct features and abilities. Virtual agents evaluate their path through the city based on empiric indicators (e.g. Level of Service (LoS), time of travel etc.) as well as secondary individual indicators (e.g. exhaustion, lucidity of the city layout, personal space available, stress etc.), which are otherwise only available with time intense and expensive surveys.

Agents are trained to prefer walking on the sidewalk and to avoid streets. The agents read the ground and understand a large range of colors. As simplified replicas of humans, the agents are primarily controlled by their visual input for collisions detection, which is an overruling factor (Gibson, 1979). A change in color of an agent is used to communicate its state to other agents, allowing typical emotions to be conveyed in situations such as when a collision occurs. Red, representing anger, or blue, representing indifference, are visible to the agents and containing information about the state of the other agent. Agents use the frequency and the amplitude of the sound to guide them towards distant locations or points of interest, which are out of sight.

Tracked output: The artificial intelligence conveys information about the effort it took the agent to reach the designated points of interest. The inherited strength an agent is provided at the beginning is reduced faster by walking upwards then moving down. The agent is also equipped to measure the time and determines if his journey was too time consuming according to the distance he had to go. The vision and distance measurements define how much personal space was available to each agent during the journey.
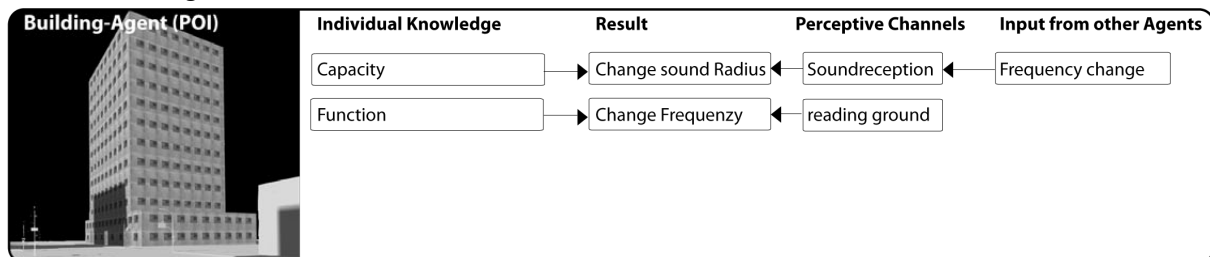
7.2. Buildings / POI

FIG. 9: Input – Output details of Building-Agents

Building-agents are provided with a position, function and capacity, all encoded in the parametric model. The building-agent is emitting a sound frequency in a defined radius. This frequency contains the information of function for other agents. When it has reached its capacity, the sound radius is damped and the frequency adjusted. The

building-agent receives the sound frequency from all other surrounding agents (e.g. pedestrians). By computing the distance to other agents, the building-agent is able to know whether these agents have crossed the door and are inside. Building-agents are also sources of pedestrian agents; according to the time of the day each building-agent is spawning other agents (e.g. pedestrians), which then start to interact with the environment.
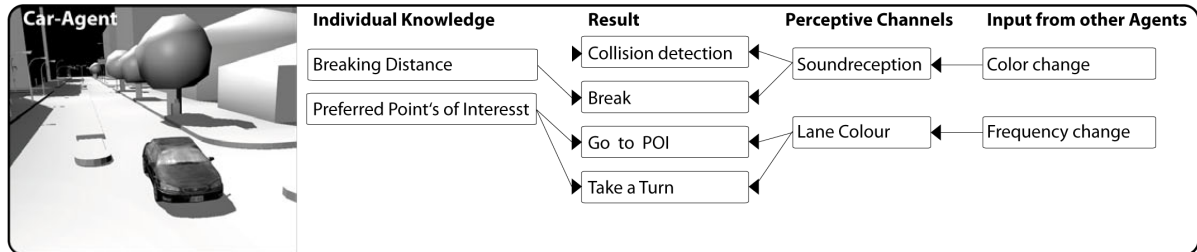
7.3. Individual Transport / Cars

FIG. 10: Input – Output details for Car-Agents

Cars are bound to the streets, which are represented by colored lanes and are automatically generated with the procedural model. At its creation, each car agent decides on its goals. Guided by the sound emissions, a car-agent is able to define its distance to each goal. With and based on its own set of preferences, then decides on which one to go to first.

Encoded in the lane is also the kind of street, implying the maximum speed for the car-agents. But the agent is also aware of all the other agents and is measuring the distance towards each of them. Pedestrian-agents are recognized when crossing streets, which leads to braking and stopping states. Other car- and public transport-agents are also monitored. Based on their locations and speed, each car-agent adjusts its speed in order to maintain a safe distances to these other vehicles. This also leads to traffic density fluctuations and congestions.

The path finding process is a 'greedy function', where the agent is trying to minimize its distance towards the goal. Because each agent is bound to the street, it can only change its path at crossing, where it decides upon arrival to turn in the direction of its goal. For ease of interaction between pedestrian-agents and car-agents, traffic lights were also incorporated, restricting one direction for the car-agents and the other for the pedestrian-agents.

Tracked output: We used the output channels to gain information about I) their absolute deviation from the maximum speed II) their accelerations and III) their deviations from a direct line to their goal. This enables us to identify streets with a high congestion risk, demanding for alternative routes or more lanes.

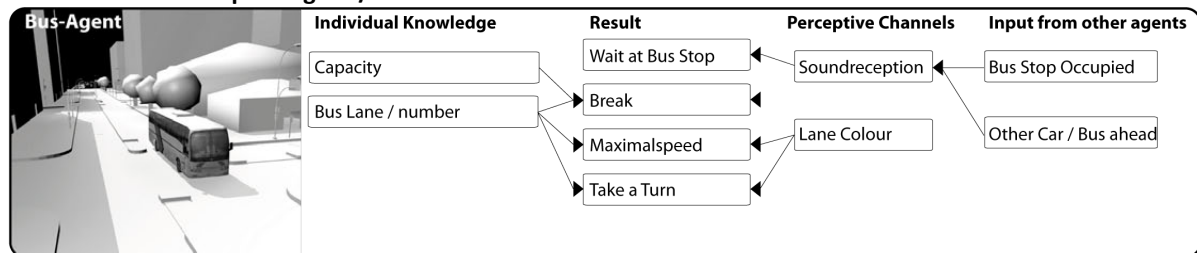## 7.4. Public Transport-Agent / Bus

FIG. 11: Input – Output details of Bus-Agents

Similar to the car-agents, public transport-agents are bound to the street but have a strict route to follow with stops. These stops are only taken if pedestrian-agents are actually present at the bus stop when the agent is arriving. (Also the speed and acceleration are different to take the motion pattern into account.)

Each public transport-agent is aware of its total and current transport capacity. Additionally, higher numbers of pedestrian-agents getting off and on at a bus stop lead to longer waiting times at the stops.

Tracked output: We were interested in the capacity level at each stop and the actual time each public transport agent needs to go from one stop to the next one.

8. Results

8.1. Output by Agent

During the simulation, every agent draws an individual color-coded line on the ground. The color change, triggered by the brain, is representative of both its emotions and its analytical experience of the environment. Every agent is entirely unique.

We have to stress that a single agent is not representative, but if a significant number of agents appear to be exhausted at the very same spot, we know that we have to adjust. Depending on the location a rest area with benches may be needed. Also, if the net travel time appears to be too long we assume that the agents would rather take public or private transport then walk. With this information we can adjust the stops of the public transportation system to increase the time efficiency for the majority of occupants.

8.2. Dübendorf Example

Dübendorf, a suburban centre of Zürich, currently houses a military airport, which will be made redundant. The area in question is the biggest to be developed in Switzerland in the near future. With an increasing trend of living in suburban areas, there is increased pressure to develop the area into a compact new centre for Dübendorf and surrounding areas. We divided the results in two parts a pedestrian based evaluation (7.1.1, 7.1.2) and a street agent interaction. Both have a similar underlying structure and represent the result of our workflow.

8.2.1. Primary Design Evolution

In this case we can show the density differences between the area of high-rise buildings in the centre and the low-density living quarters on the outskirts. There is no absolute rule about density or traffic - for some attractions an increased flow of occupants is favored, while for others this is not the case. We show in figure 12 the layer of personal space. The problem we identified here was that the entrances / exits of the buildings were opposite to each other. They would be better placed in an asymmetric way to avoid collisions.
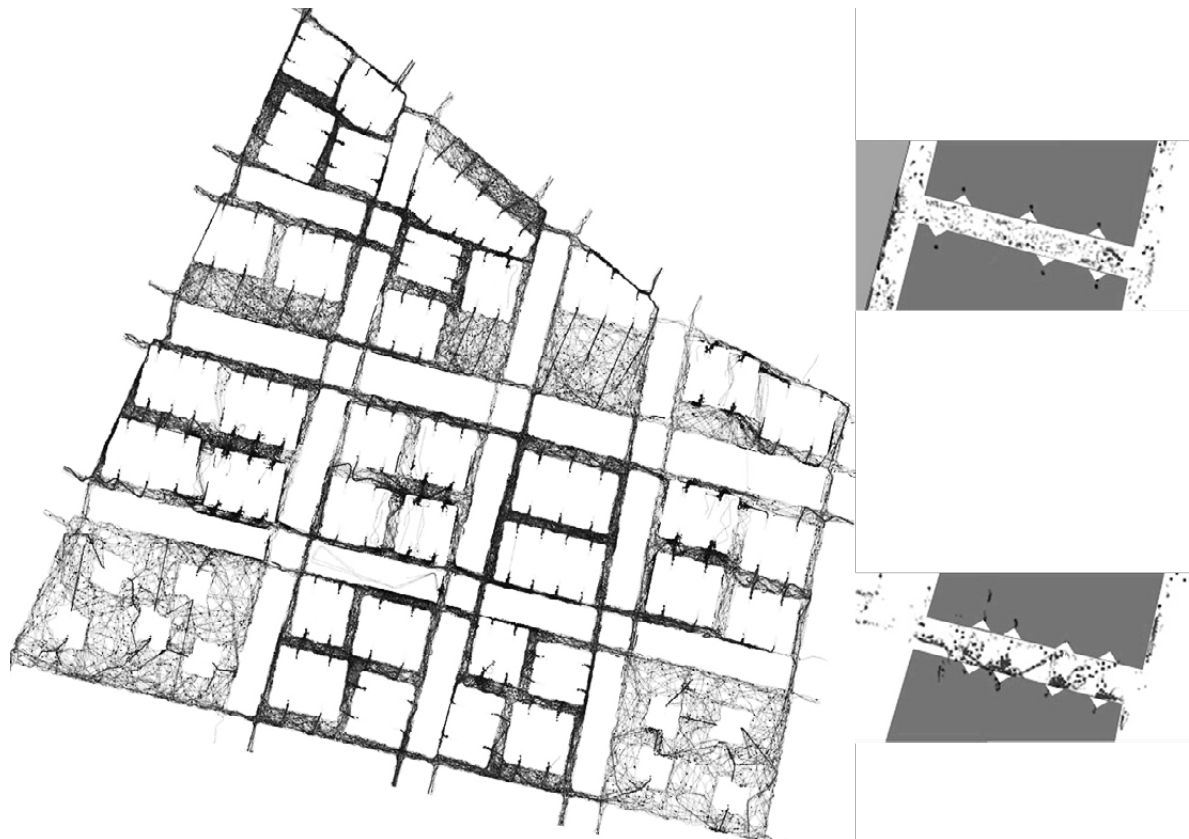

FIG. 12: Left: Quantified visualization of 6800 agents within the given area. Middle and right: Occupant stress analysis; dark dots represent not enough individual space.

To handle the amount of agents in this case, we group agents into three major groups (Jager et al., 2001). The group which is coming from home (GotoWork), the group which comes from work (GotoHome) and the group which has no specific goal to reach and occupy the streets as a present background noise.

8.2.2. Secondary Design Evolution

The setup consists of a pedestrian crossing, with different typical activities, and a street, which represents an obstacle to the agents. This time the agents find their way from their source to their final destination by making a selection from a list of possible secondary goals and marking their way within their individual path. As a result we can see increased traffic around the high-rise buildings as well as the crossing of the streets in the north (FIG. 13). As seen in the collision map, there is no stress problem at street crossing. However, even with the same amount of pedestrians, there is a problem in the middle of the pedestrian walk way. This is due to several reasons. One reason is the agents' flocking behavior, making the agents align themselves there, after increasing the amount of iterations. Additionally, if an agent is making decisions in that particular area it is likely that he will change his goals. Interestingly, the agents do not favor the two asymmetric pedestrian crossings. In order to find the next crossing, they can easily block each other's path. The total-time-spent map reveals the need for another public traffic stop at the south end to even out the time needed to reach all goals.
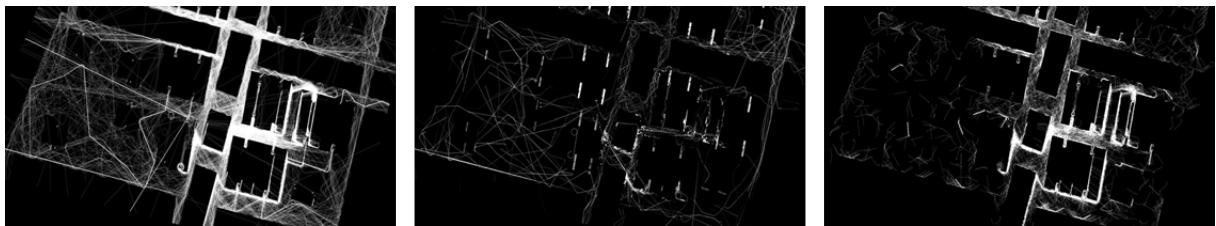

Fig. 13: High-density crossing; from left to right: Density, collision, and time map.

8.2.3. Third Design Evolution

In this case we focus on the evaluation of the street bound agents. We extracted only the maps drawn by the cars or buses. In this case we also would like to show the difference over the day. The case consists of two areas, to the west, there are bigger volumes with work functions and corresponding secondary functions, such as cafeteria etc.; in the east, there are residential buildings with lower height and capacity.
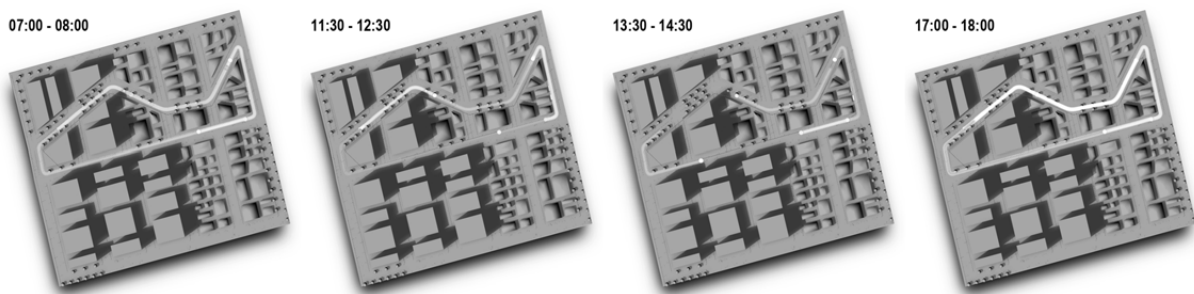

FIG. 14: Time-depended use of the Bus

In this case we show a single bus line and the utilization of its capacity (FIG. 14). The bus is turning clockwise and absorbs agents traveling for a longer distance then walking. In general the morning (07:00 – 08:00) and evening (17:00 – 18:00) simulations shows a higher exploitation over the whole loop.
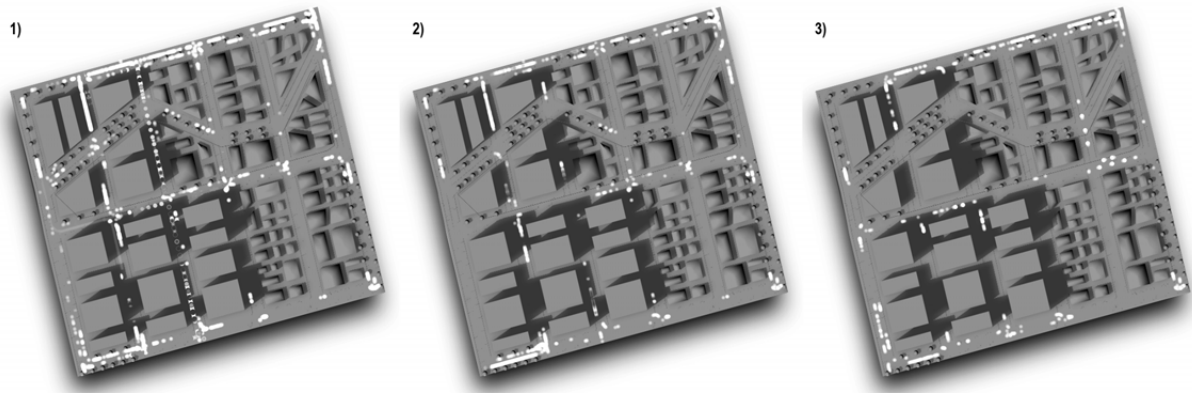
FIG. 15: Congestion effects of cars in the morning traffic over time

In this case we se how the morning traffic is dissolving over 2 hours (FIG. 15). These images are part of a dynamic movie showing the density fluctuations. In Fig. 16 the same agents are also showing where the path they took deviates from an optimal line.
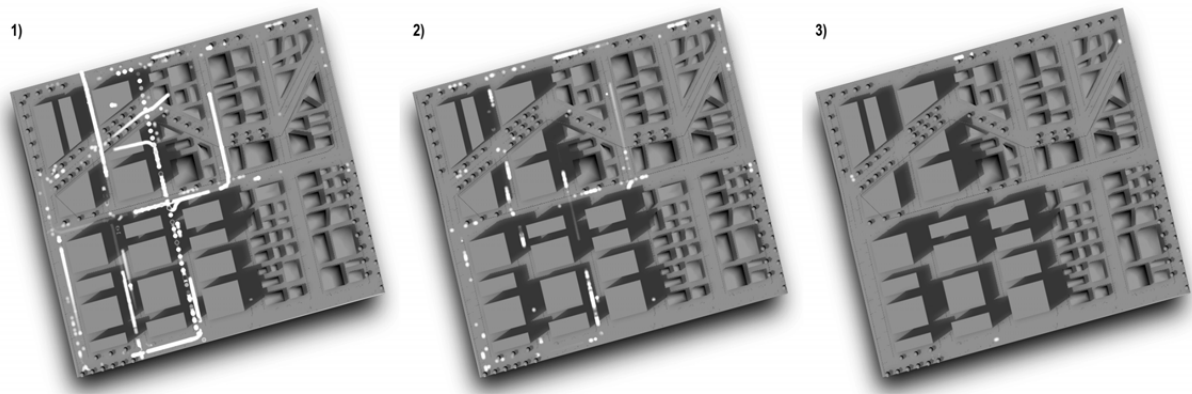

FIG. 16: Deviation from optimal path in morning traffic over time

9. Impact on Planning Decisions

Our simulation method enables the urban planner to adjust the design from small scale to large scale. Each evaluation, sorted by kind of agent, has different planning implications.

Pedestrian-Agent: The density maps have implications on the functional layer of the city – commercial function welcomes a higher rate of pedestrian traffic but residential function might move out of the area. The collision map shows where the stress level for pedestrians is increased and where the sidewalks have to get wider. The time-map has an impact on another system, which defines the position of the public transports stops.

Bus-Agent: The evaluation by the bus agents has to be looked at over time. There, the congestion effect within the bus-system is actually visible. Also the efficiency, the ability to transport pedestrians, of the system is varying for each individual bus and has implications to provide additional buses for a specific time and part of the bus system.

Car-Agent: The most profound part of a city is its street network – slow changing. Therefore an initial setting of the streets has to be done with a lot of care. But the knowledge of congestion effects also allows redirecting traffic in an existing configuration prior to the change, e.g. construction etc. .

With a limited amount of resources, this allows the urban planer to choose the smallest possible intervention to gain the largest output. Small obvious changes are more likely to be accepted by the occupants and therefore have a higher chance of creating a permanent effect on the city. In the design stage changes can be simulated and evaluated. These evaluations might lead to contradicting needs allowing stakeholders to make a decision reflecting their opinion.

The high-resolution pictures and movies (Fig. 14) are then a method of communicating to all stakeholders, allowing a discussion about different design ideas and address problems. With an increased number of decisions made unilaterally, a clear communication reduces time and meeting requirements.

FIG. 17: Detailed graphical output for communication

10. Discussion

The results, divided according to each agent , are a useful tool and allows to discuss the implications between different field, such as urban planners, traffic planers as well as politicians. Even with useful results that are relatively easy to understand, we have to research possible deviations between our prediction and reality. Despite reliable statistics, it is virtually impossible to reconstruct the decision process of human beings. For this reason we concentrate on the path finding abilities of the agents and rely on predefined goals. Another limitation of our method is the calculation time. The enhanced complexity of the urban layout increases the calculation time exponentially.

11. Conclusion

This project included for the first time a dynamic environment, connecting the physical environment to the movement und utilization patterns of its occupants. This is a crucial part to understand better the movement of extremely large crowds within a city. This dynamic environment guides agents to POI's they would have chosen from the start - but resembles a more natural path finding process, where the path and the goal get adjusted according to the environment constantly. The behavior patterns of pedestrians and their decision processes on the small scale affect urban users on all scales. This increased knowledge about the behavior, the interaction between actors in a city and movement patterns are on an unprecedented scale enabling planners and decision makers to find more subtle ways to adjust the urban fabric and offer the opportunity to investigate parts of a city and to optimize corresponding aspects with minimal interventions on the urban level.

This attempt of bridging the gap between function, behavior and structure is still only automated from structure to function and hands over the behavior model to a different system. Therefore, an automatic iteration of the loop is still far from being an integrated, interactive system. A closer integration of function and behavior reduces calculation time and will lead to a practical integrating.

12. Acknowledgment

13. References

Arnott R. und Small K.A.: 1998, Urban Spatial Structure, journal of Economics Literature, Vol. XXXVI, Pages: 1426 - 1464

Aschwanden G, Halatsch J, Schmitt G, 2008, Crowd Simulation for Urban Planning, eCAADe proceedings, Antwerpen, 45-82

Batty M, Xie Y, 1994, Modeling inside GIS: Part 2. Selecting and calibrating urban models using ARC-INFO, International Journal of Geographical Information Science, 8, 451-470

LeBon, G.: 1895, Psychologie des Foules, Alcan, Paris.Brooks RA, 1991, Behaviour-based humanoid robotics.

Diener, R., Herzog, J., Meili, M.,de Meuron, P., Schmid, Ch, (2006). "Switzerland - An urban Portrait." ETH Studio Basel: Contemporary City Institute, Pages: 64-109.

Egges A., Kshirsagar S., Magnenat-Thalmann N.: Generic Personality and Emotion Simulation for Conversational Agents

Gibson J J, 1979 The Ecological Approach to Visual Perception (Houghton Mifflin, Boston, MA)

Guy S., Chhugani J., Kim C., Satish N., Lin M.C., Manocha D., and Dubey P., Clearpath: Highly parallel collision avoidance for multi-agent simulation. Proc. Of ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pages 177-187, 2009.

Halatsch, J., Kunze, A., Burkhard, R., and Schmitt, G. (2008a). ETH Value Lab - A Framework For Managing Large-Scale Urban Projects, 7th China Urban Housing Conference, Faculty of Architecture and Urban Planning, Chongqing University, Chongqing.

Halatsch, J., Kunze, A., and Schmitt, G. (2008b). Using Shape Grammars for Master Planning, Third conference on design computing and cognition (DCC08), Atlanta.

Helbing D., Molnar P., Social force model for pedestrian dynamics. Physical Review E, 51:4282, May 1995. © 2008 The Americn Physical Society

Hillier B, 2002, A theory of the city as object: or, how spatial laws mediate the social construction of urban space, Urban Design International, 7, 3, 153-179

Jager, W., Popping, R. and van de Sande, H.: 2001, Clustering and Fighting in Two-party Crowds: Simulating the Approach-avoidance Conflict, Journal of Artificial Societies and Social Simulation, 4(3).

Magnenat-Thalmann, N. and Thalmann, D. (eds.): 2004, Handbook of virtual humans, Wiley, Chichester.

Müller P., Wonka P., Haegler S., Ulmer A. and Van Gool L. 2006. Procedural Modeling of Buildings. In Proceedings of ACM SIGGRAPH 2006 / ACM Transactions on Graphics (TOG), ACM Press, Vol. 25, No. 3, pages 614-623.

Musse, S. R., Thalmann, D.: 2001, Hierarchical Model for Real Time Simulation of Virtual Human Crowds, IEEE Trans. on Visualization & Computer Graphics, 7(2), pp. 152-164.

Narain R., Golas A., Curtis S., and Lin M. C., Aggregated dynamics for dense crowd simulation. ACM Transactions on Graphics (Proc. Of ACM Siggraph Asia), 2009

Parish Y, Müller P, 2001, Procedural modeling of cities. ACM, 301-308

Parish Y. I. H. and Müller P. 2001. Procedural Modeling of Cities. In Proceedings of ACM SIGGRAPH 2001, ACM Press / ACM SIGGRAPH, New York. E. Fiume (ed), COMPUTER GRAPHICS, Annual Conference Series, ACM, pages 301-308.

Penn + Turner 2001 From isovists to visibility graphs: a methodology for the analysis of

Penn, A. and Turner, A.: 2001, Space Syntax Based Agent Simulation, in M. Schreckenberg, S. D. Sharma (eds.), Pedestrian and Evacuation Dynamics, Springer-Verlag, Berlin.

Reynolds C. W., Flocks, herds and schools: A distributed behavior model. ACM SIGGRAPH Computer Graphics, 21:25-34, 1987

Reynolds C. W., Steering behavior for autonomous characters. Game Developers Conference, 199

Pelechano N., Allbeck J. M.,and Badler N. I., Virtual Crowds: Methods, Simulation and Control. Morgan and Claypool Publishers, 2008.

Procedural Inc., Zurich, Switzerland; http://www.procedural.com

Surowiecki, J., Silverman, MP., 2007. The wisdom of crowds. American Journal of Physics, Volume 75 p. 1904

Ulmer A, Halatsch J, Kunze A, Muller P, Van Gool L, Incorporated P, Zurich S, 2007, Procedural design of urban open spaces, vol 25, 351-358

Still, G. K.: 2000, Crowd Dynamics, PhD thesis, Warwick University.

Treuille A., Cooper S., and Popovic Z., Continuum crowds. Proc. Of ACM Siggraph, pages 1160 – 1168, 2006

Umeda Y, Takeda H, Tomiyama T, Yoshikawa H, 1990, "Function, Behavior and Structure" in Applications of Artificial Intelligence in Engineering V, Computational Mechanics Publications/Springer Verlag, 177-193

Tecchia, F. and Chrysanthou, Y.: 2000, Real-Time Rendering of Densely Populated Urban Environments, in Proc. Eurographics Rendering Workshop.

UN-Habitat, 2009, Planning sustainable cities: global report on human settlements 2009, United Nations, Human Settlements Programme, http://www.unhabitat.org/downloads/docs/GRHS2009/GRHS.2009.pdf

Van den Berg, J., Seawall,. Lin M.C., and Manocha D., Virtualize d traffic: Reconstructing traffic flows from discrete spatio-temporal data-temporal data. Proc. Of IEEE Virtual Reality Animation, 2004.

Van de Berg, Guy Stephan J., Lin M., and Manocha D., Reciprocal n-body collision avoidance. In International Symposium of Robotics Research, 2009

Vanegas C, Aliaga D, Beneö B, Waddell P, 2009b, Visualization of simulated urban spaces: Inferring parameterized generation of streets, parcels, and aerial imagery, Ieee Transactions on Visualization and Computer Graphics, 424-435

Vanegas CA, Aliaga DG, Benes B, Waddell P, 2009a, Interactive Design of Urban Spaces using Geometrical and Behavioral Modeling, ACM Transactions on Graphics (also in Proceedings SIGGRAPH Asia), 28(5)

Waddell P, 2002, Modeling urban development for land use, transportation, and environmental planning, Journal of the American Planning Association, 68, 297-314

Waddell P, Ulfarsson G, 2004, Introduction to urban simulation: Design and development of operational models, Handbook of transport geography and spatial systems, 5, 25-78

Wegener M, 1994, Operational urban models state of the art, Journal of the American Planning Association, 60, 17-29

Wu F, Webster, CJ, 1998, Simulation of natural land use zoning under freemarket and incremental development control regimes, Computers, Environment and Urban Systems, V. 22 Issue 3 p. 241-256

Zipf G. K., Human behavior and the principle of least effort. Addison-Wesley Press, 1949