BAUHAUS-UNIVERSITÄT WEIMAR

# Procedural City Layout Generation Based on Urban Land Use Models

by

Saskia A. Groenewegen

A thesis submitted in partial fulfillment for the degree of
Master of Science in Media Systems

in the
Faculty of Media
Department of Media Systems

Thesis Supervisors:
Prof. Dr. Bernd Fröhlich, Bauhaus-Universität Weimar
Dr. Ir. Rafael Bidarra, Delft University of Technology
Ir. Ruben M. Smelik, TNO Defense, Security and Safety

22nd March, 2009

# Declaration of Authorship

I, Saskia A. Groenewegen, declare that this thesis titled, “Procedural City Layout Generation Based on Urban Land Use Models” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a M.Sc. degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

# *Abstract*

Training and simulation applications in virtual worlds require significant amounts of structurally plausible urban environments. Procedural generation is an efficient way to create such models. Existing approaches for procedural modelling of cities aim at facilitating the work of urban planners and artists, but either require expert knowledge or external input data to generate results that resemble real-life cities, or they have long computation times, and thus are unsuitable for non-experts such as training instructors. We propose a method that procedurally creates layouts of cities from high-level, intuitive user input such as city size, location and historic background. The resulting layouts consist of different kinds of city districts which are arranged using constraints derived from established models of urban land use. Our approach avoids the need for external expert engagement in the creation process and allows for the generation of large city layouts in seconds, making it significantly faster than comparable agent-based software and thus supporting the needs of non-expert creators of virtual cities for many applications.

# *Acknowledgements*

This thesis would not have been possible without Professor Rafael Bidarra of the Delft University of Technology, who welcomed me into the TU Delft Computer Graphics Group. I thank him for the corrections and revisions made to my drafts and for providing a supportive and creative environment that it was a pleasure to work in.

I thank Ruben Smelik, who supervised me at the TNO Modelling and Simulation department in Den Haag, who I could always turn to with my questions about programming and who diligently read everything I wrote and offered many helpful comments. No one has put more time and effort into supporting me during this thesis than he has.

A thank you also to Professor Bernd Fröhlich, who kindly guided me through this thesis and whose ability to always spot every weak point was immensely helpful to me.

I thank James Casavant and Michiel Bouwhuis for proofreading and trying to spot all my English grammar mistakes, despite the fact that they were tremendously busy themselves.

Megan Ng, Egbert Stolk and José Beirão were the architects who helped me with their expertise in urban design by checking all the assumptions I made on their subject and providing me with additional literature, I thank them very much for their time and their help.

I would also like to thank everybody at TNO Modelling and Simulation for supporting me, especially Klaas-Jan de Kraker, who enabled me to work there and helped with revising the paper that was written on my thesis topic.

All my fellow TU Delft students on the 12th floor from the Computer Graphics and the Man Machine Interaction groups. Thank you for all the inspiration, the coffee and the cake.

The people at the Virtual Reality and the Augmented Reality labs in Weimar for the many long hours spent together on project work in the labs and on coffee breaks at the M18 café during my studies, and all others who made life in Weimar what it was: Stefanie H., Marco, Sylvia, Gordon, Anselm, Sebastian, Uwe, Daniel K., Alex, Petro, Daniel D., Kristin, Stefanie Z., Tobias, Ferry, Thommy, Christian P., Daisuke and Christian N., you were a constant source of fun and inspiration. Also my former flatmates in Weimar, without whom life would not have been the same: Sanne, Mathias, Felix, Armin, Marie and Annegret.

And finally, I would like to thank all my extended family and friends who encouraged and supported me throughout my studies, most of all my mother. Thank you all.

# Contents

# List of Figures

# List of Tables

*To Margareta*

# Chapter 1

# Introduction

Virtual worlds have become an ubiquitous presence in the lives of many people. Many are encountered in entertainment: There are the computer games like World of Warcraft or Oblivion, with their vast terrains, complex cities and thick forests that users can spend months exploring. There are also the simulations that emulate the real world, like Second Life and others that mimic the parallel reality of the “Metaverse”. The other big area where virtual worlds are used is in training and simulation: Whether it is airplane pilots learning to fly in a simulator or soldiers training for missions between the virtual houses of a remote country, virtual environments offer a cheap alternative or addition to real world training.

What all those scenarios have in common is that they need content. Someone has to build every tree in World of Warcraft and every house in a city simulation. And because repeating scenarios are at best not engaging and at worst make training inefficient because trainees know what is coming, new content has to be created constantly. This is the reason why procedural modelling is employed. Procedural modelling is defined as the creation of three-dimensional models, textures and other objects - even music - by applying algorithms rather than shaping objects manually. It is applied in the case when modelling by hand would be too cumbersome for the artists who usually build the objects, or if it would be an indomitable amount of work. Imagine having to model every tree in a forest by hand!

An interesting area of procedural modelling is the generation of cities. This has already been done with impressive results to regenerate archaeological sites like ancient Rome or Pompeii, huge cities whose manual recreation in minuscule detail would have been impossible. Those recreations are based on large amounts of input data from which the city is then built: street maps, population density plans, footprints of the city, maps that describe the land use in various parts of the city, and other data serve as input to steer the procedural generation process. This means two things: one the one hand, the city will be very true to the intended results and look most impressive. On the other hand, a vast amount of knowledge has to be fed into the system before a realistic result is produced.

Now we imagine how this technique could be used in another setting. An instructor has to teach a number of students tactical maneuvers in an urban environment, for example rescuing people in an emergency. The students have to attune their behaviour and reactions to the given environment. This means that the training cannot take place in the same urban environment every time, new fictional cities have to be created regularly. The instructor, however, is no urban planner and not familiar with the structures of cities and models. She would need a software that integrates procedural generation with the knowledge of how a city is shaped, where certain kinds of buildings are located and how the landscape in which the city is build influences those factors. If she had a program with these abilities, she could generate new urban environments for her training simulations on the fly, provide her students with new, challenging situations and concentrate on her actual job as instructor.

Such a program - or the necessary algorithms - is not available yet, as we found out by surveying the existing approaches to automatic city generation. It is the focus of this thesis: to find a way to create a city with a believable layout that exhibits the same structure as cities found in the real world, and to do so in a very short time. To this end, we will look at models which describe the spatial layout of cities in different parts of the world. We then develop an algorithm to integrate those land use models into a procedural modelling framework to generate the layout of a city. Finally, we will evaluate the results that our implementation produces with the help of urban planning experts and see that they are plausible, that generation times are fast and that the software can be considered a significant step forward for any application that needs plausible city models.

# Chapter 2

# State of the Art in Procedural City Modelling

Figure 2.1: Examples for procedural world generation in games. Left: fully procedural world in Darwinia (2005). Center: procedural plants in The Elder Scrolls IV: Oblivion (2006). Right: procedural landscape in Infinity: The Quest for Earth (2008)

In the area of computer graphics there are many applications for large virtual environments. Examples of which include computer simulations for (military) training and computer games which derive part of their appeal from the ability to explore vast terrains. Creating large landscapes by hand is a tedious task for modellers and artists, thus the ability to automatically generate them is much appreciated. Procedural modelling deals with the automatic creation of models and textures from sets of rules. Given input parameters, algorithms are applied to generate a scene according to the rule set. Examples for procedural modelling techniques include L-Systems [Prusinkiewicz and Lindenmayer 1991], fractals [Mandelbrot 1982] and shape grammars [Stiny 1975]. Several games already use procedural content for terrain and vegetation creation or level layout. Examples include Diablo [Blizzard Entertainment 1996], Morrowind and Oblivion [Bethesda Softworks 2002, 2006], Darwinia [Introversion Software 2005], Infinity [Brebion 2009] and Spore [Maxis 2008]. Figure 2.1 shows examples for procedurally generated game content.

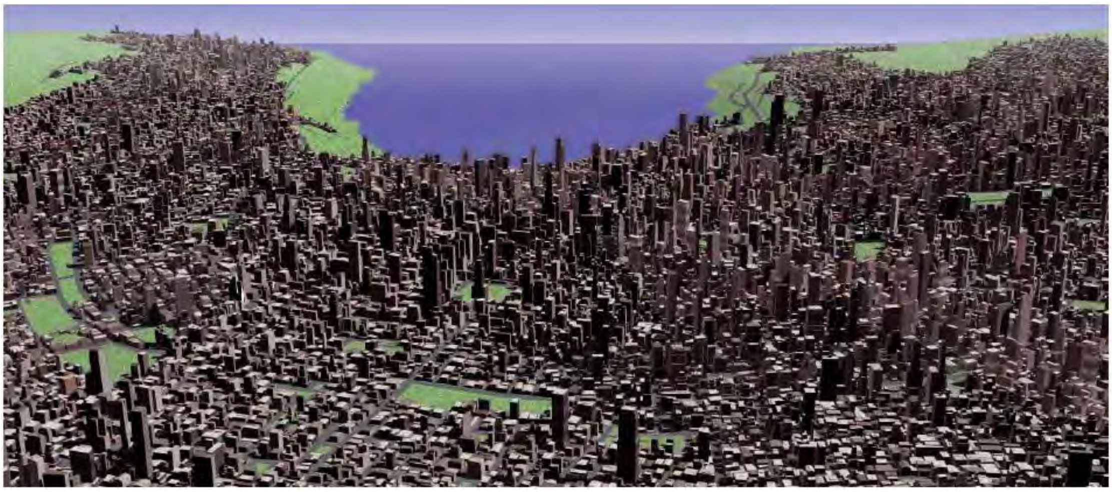


FIGURE 2.2: A procedurally generated city [Parish and Müller 2001]

An important area within the field of procedural modelling is the generation of urban landscapes. Procedural modelling of cities mainly deals with three objectives: creation of road networks, creation of buildings (including the generation of interiors and the production of façades) and land use within a city. The following sections will give an overview over techniques for each of those areas.

## 2.1 Road Network Generation

The generation of road networks can be done using a variety of methods: Templates, Voronoi Diagrams, L-Systems, Tensor Fields, and Agents.

### 2.1.1 Templates

The most common method to create roads is through the use of templates, as suggested by [Parish and Müller 2001] and [Sun et al. 2002]. They note several frequent patterns ("the internal regularity underlying the diversity of appearances" [Sun et al. 2002]) in road networks that they aim to reconstruct. Figure 2.3 shows different patterns for street layouts.

A template to create a road pattern consists of two parts: rules that define the overall structure and parameters that provide different input. The population-based template uses a Voronoi diagram of the density points of a population density map as a rule, creating the roads from the edges of the Voronoi diagram. The raster and radial templates include rules for the iterative expansion of streets: The algorithm starts from a single point and creates streets by generating vertices according to the used template until a

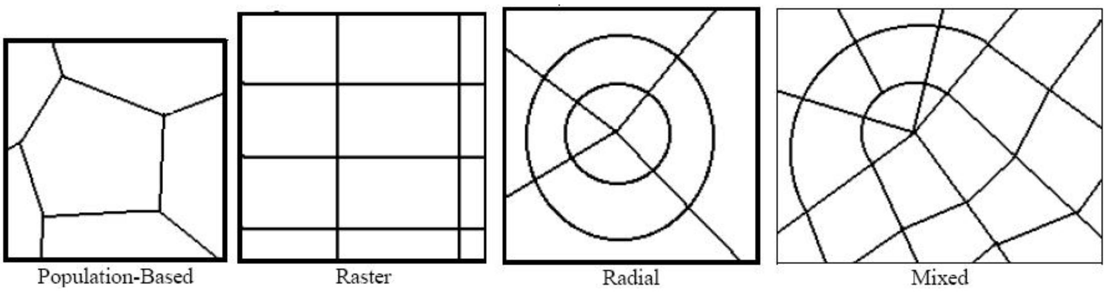


FIGURE 2.3: Road patterns [Sun et al. 2002]

bounding box is reached.

Street generation can be influenced by factors such as terrain elevation (cf. Figure 2.4) or population density. It should be noted that only the bigger streets, or highways, are created using patterns. Smaller streets are inserted into the remaining areas as simple grids.

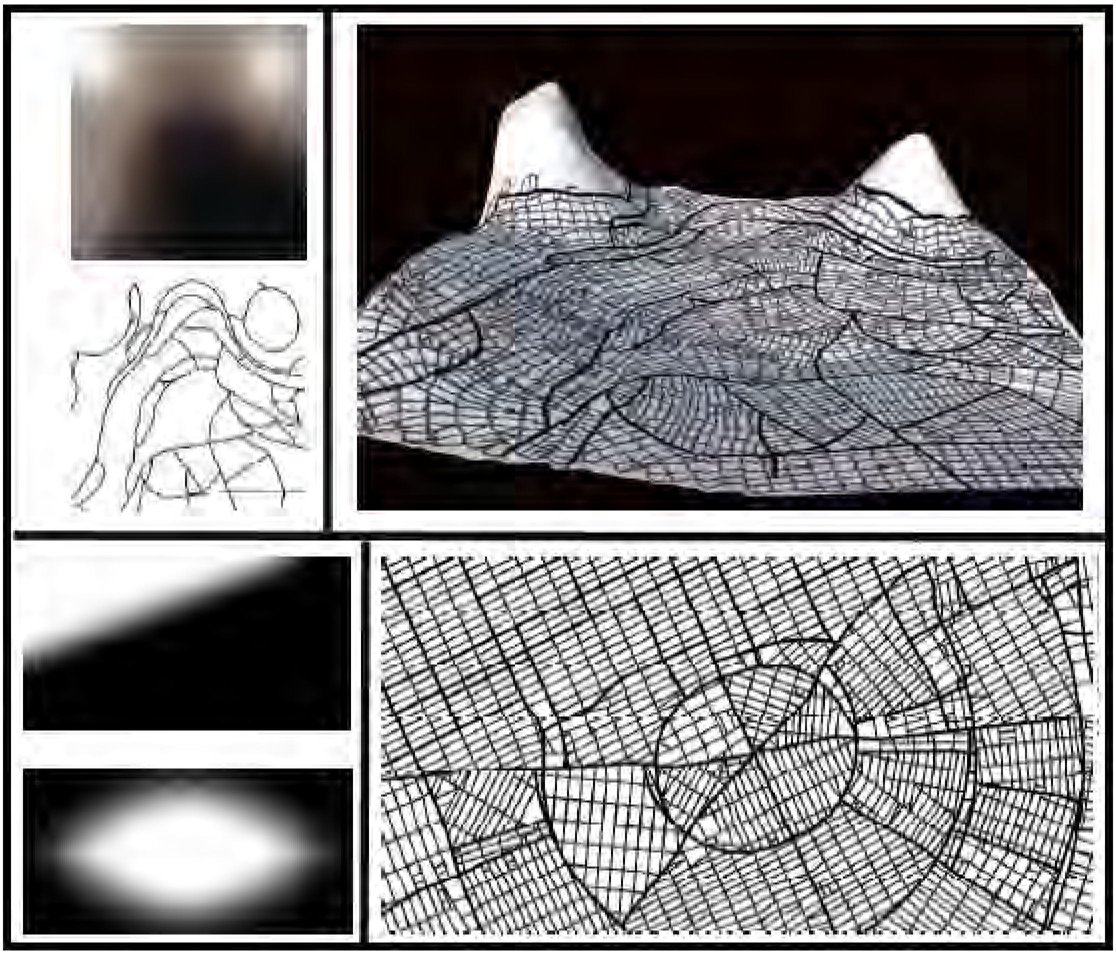


FIGURE 2.4: Top: streets created with an elevation map as input parameter. Bottom: two different street pattern control maps (they control which pattern is used where) and the resulting street layout. [Parish and Müller 2001]

### 2.1.2 Voronoi Diagrams and L-Systems

[Glass et al. 2006] use a combination of Voronoi diagrams and L-Systems as basis for street generation, having found visual similarities between those structures and the street layout in real-world settlements. Using a similar approach to the one described in Section 2.1.1, main roads are created from a Voronoi diagram whose seeds are determined via a population density map. Smaller streets are grown in the areas between main roads using an L-System grammar (cf. Figure 2.5).

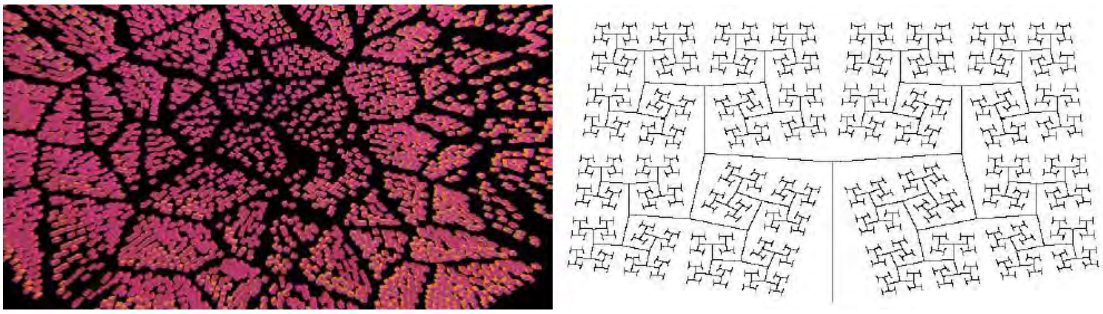


FIGURE 2.5: Left: distribution of main roads based on a Voronoi diagram. Right: smaller streets created with an L-System grammar. [Glass et al. 2006]

### 2.1.3 Tensor Fields

Another approach to street generation is presented by [Esch et al. 2007] and [Chen et al. 2008], who use user-guided tensor fields as input (shown in Figure 2.6). The basic idea is that streamlines are traced from seed points in the major eigenvector direction until a stopping condition is met. New seed points are then placed along those traced lines to create roads in the perpendicular (minor eigenvector) direction.

### 2.1.4 Agents

A very different approach to street generation is the use of an agent-based behavioural model, described in [Lechner et al. 2003], [Lechner et al. 2006] and [Watson 2007] and shown in Figure 2.7. This method is based on the idea that the previously described models depend too much on input data and that L-Systems grow overly complex when trying to cover many different cases. The only input necessary for this approach is a terrain description. Different agents with different simple behavioural rule sets are used, the idea being that complex behaviour will emerge from their local interaction. Two types of agents are used: extenders and connectors. Extenders roam the unconnected area and propose new roads, while connectors walk on the roads and build connections between roads when a nearby road is hard to reach. Constraints limit the emerging

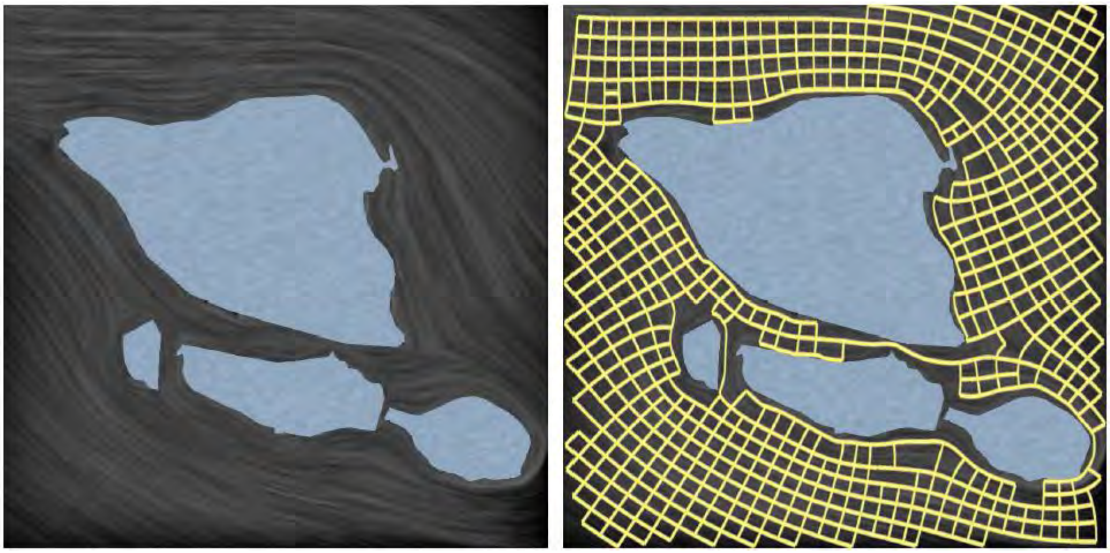


FIGURE 2.6: Tensor fields as input for street generation [Esch et al. 2007]

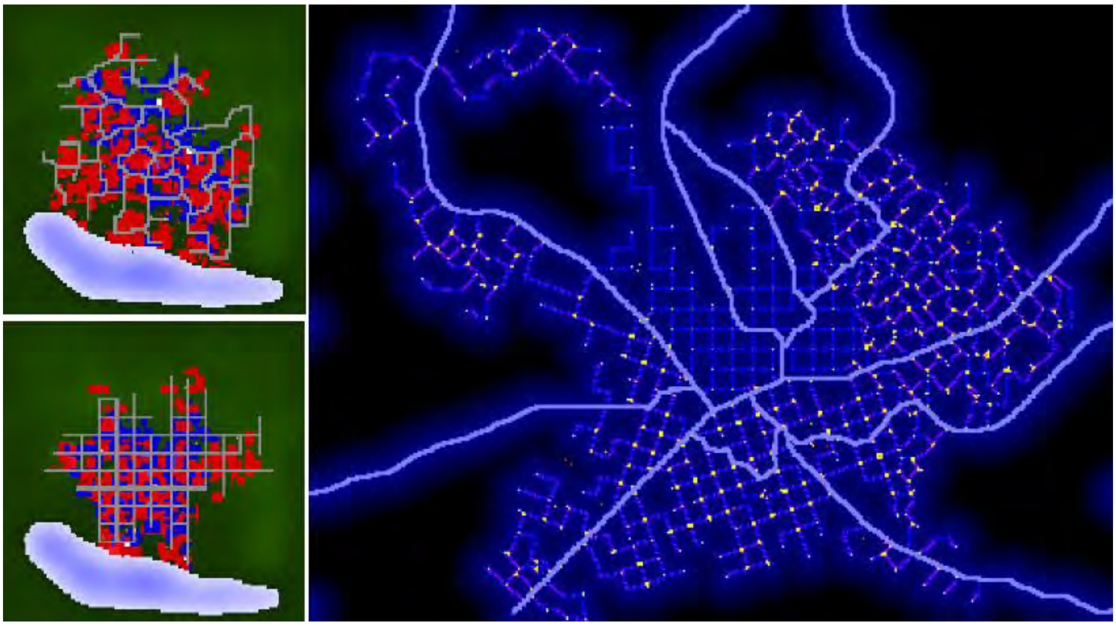


FIGURE 2.7: Road generation with agents. Left: road map created by agents using different constraints [Lechner et al. 2003]. Right: refined agent behaviour to create different patterns and road sizes in one map [Watson 2007]

grid pattern (from very linear to organic). This approach creates no main roads, only small roads, and it is therefore arguable whether the created network looks believable. A refined approach was introduced in [Lechner et al. 2006] and [Watson 2007], creating highways, smaller streets and different street patterns. While producing good looking results, the disadvantage of this approach is the time it takes to build a road map (30 minutes for a simple map in [Lechner et al. 2003]) and the need to program the agents in a realistic way.

## 2.2 Building Generation

Procedural building generation mostly produces Manhattan-like assemblies of skyscrapers set into a network of highways and smaller streets, as described by [Parish and Müller 2001], [Greuter et al. 2003], [Laycock and Day 2003], [Wonka et al. 2003], [König and Bauriedel 2004], [Greuter et al. 2004], [Coelho et al. 2005], [Mcbryde 2005] and [Kelly and Mccabe 2007]. Examples can be seen in Figure 2.8.

Attempts at modelling smaller houses exist especially in the context of preserving cultural heritage: modelling vernacular houses [Birch et al. 2001, Yong et al. 2004], recreating ancient Pompeii [Müller et al. 2006], or rebuilding Rome [Frischer 2008]. [Müller et al. 2006] expand a shape grammar originally designed for office buildings to create residential homes of a wealthy suburb. Examples are shown in Figure 2.9.

Building shapes are generated using various approaches, introduced in these following sections.

### 2.2.1 Floor Plan Extrusion

[Greuter et al. 2003] combine geometric primitives to form floor plans for buildings. Each building section is constructed by extruding a floor plan into a three-dimensional shape (Figure 2.10). The resulting skyscraper buildings are used in an application mainly concerned with computationally efficient real-time rendering and less with detail or realism.

### 2.2.2 L-Systems

[Parish and Müller 2001] employ L-systems to model buildings out of rectangular volumes (Figure 2.11). Their method utilizes a restrictive “decreasing apices” L-system class which uses the bounding box of a building as axiom and refines the geometry with

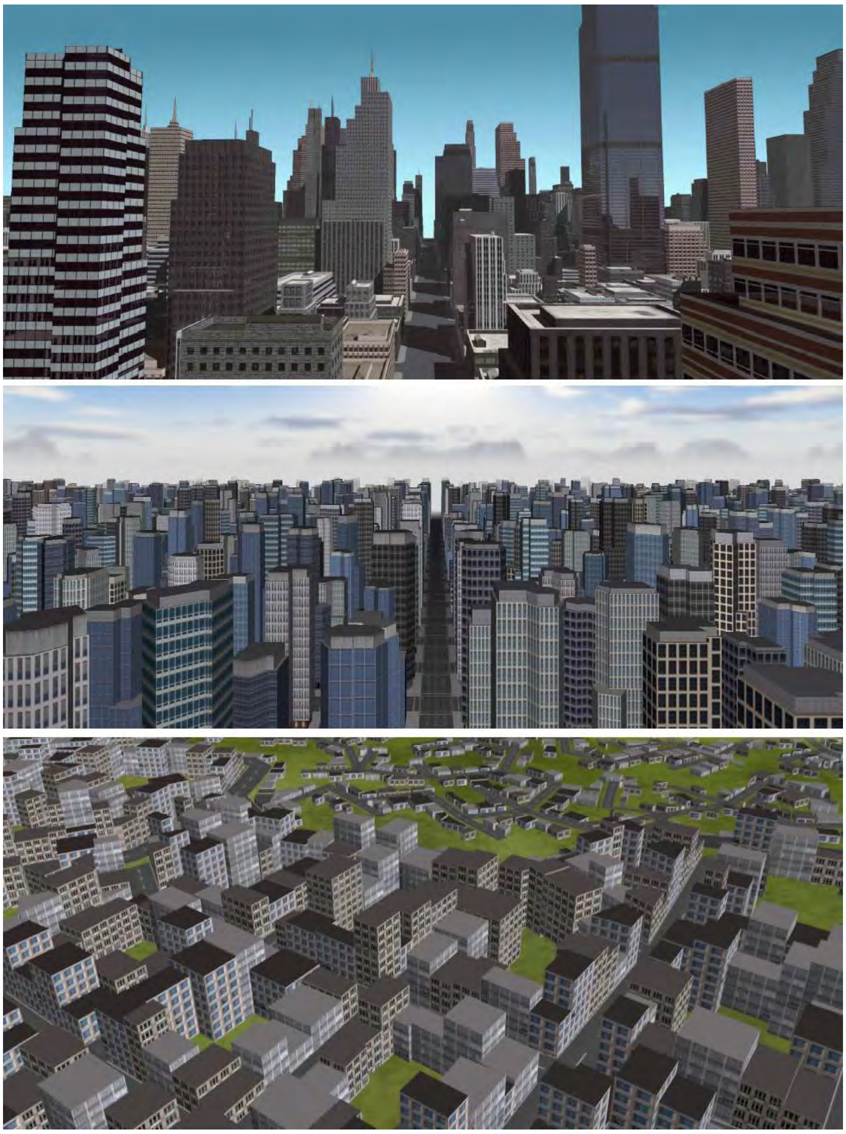


FIGURE 2.8: Three examples for Manhattan-like procedural cities, from [Parish and Müller 2001], [Greuter et al. 2003] and [Kelly and Mccabe 2007]

each iteration. This approach is useful for modelling large numbers of relatively simple buildings.

### 2.2.3 Split Grammar

[Wonka et al. 2003] introduce the idea of the "split grammar", a formal context-free grammar designed to produce realistic buildings. The split grammar resembles an L-system but uses shapes as primitive elements rather than letters or symbols. Shapes are replaced by one or more other shapes according to production rules. The generation starts with an initial shape and proceeds recursively until only terminal shapes remain. Figure 2.12 illustrates a split grammar and exemplary results.

FIGURE 2.9: Examples for procedurally generated residential buildings. Top: vernacular houses in China [Yong et al. 2004]. Center: recreation of ancient Pompeii [Müller et al. 2006]. Bottom: wealthy suburbia model [Müller et al. 2006].

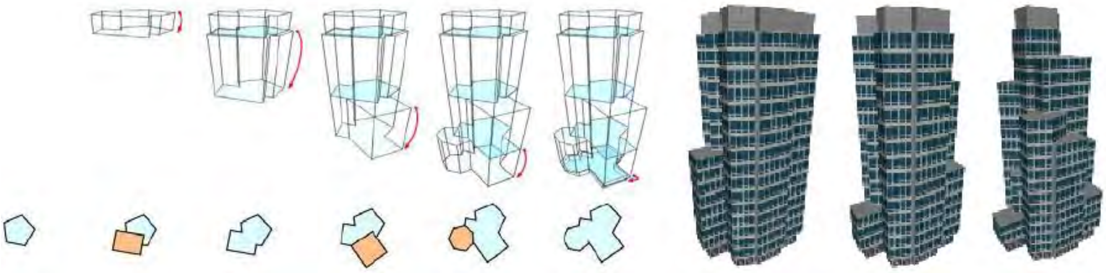


FIGURE 2.10: Building generation through merging and extruding primitive shapes [Greuter et al. 2003]

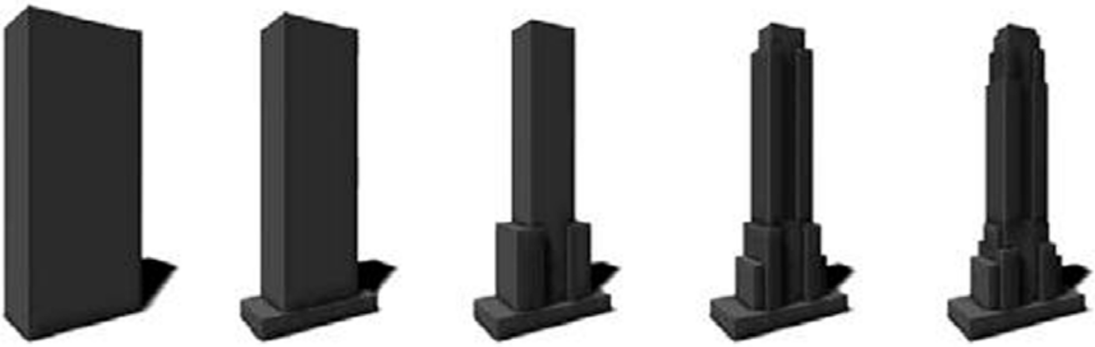


FIGURE 2.11: Building creation using L-systems in [Parish and Müller 2001]

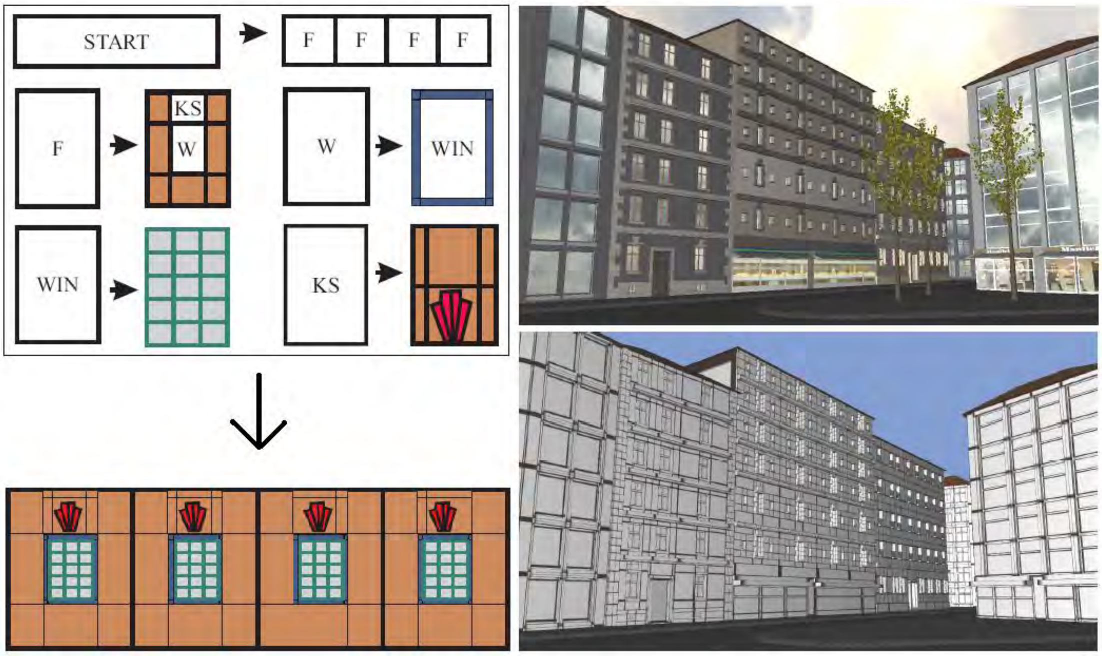


FIGURE 2.12: Top left: rules for a split grammar. Bottom left: result of the deviation of the grammar. Top right: several buildings generated using split grammars with the terminal shapes rendered as little boxes. Bottom right: the same scene without textures to show the underlying boxes. [Wonka et al. 2003]

### 2.2.4 Shape Grammar

[Müller et al. 2006] expand on the idea of split grammars by creating a shape grammar to model large, detailed urban environments. The difference between split grammars and shape grammars consists mainly in the use of context-sensitive (instead of context-free) shape rules, and in this case allows the possibility to also model roofs and rotated shapes. A vocabulary of basic shapes is used. The grammar is able to not only model business buildings but also suburban homes, as seen at the bottom of Figure 2.9.

## 2.3 Land Use Generation

Most procedural cities lack a convincing model for land use. Usually a procedurally generated city consists of only one kind of building and no distinction is made between different districts, such as commercial and residential areas, much less does it utilize a land use model that is based on real urban geography.

The existing approaches to create a city with different districts by [Lechner et al. 2003], [Lechner et al. 2004] and [Lechner et al. 2006] rely on agents to divide land into commercial and residential regions, which is shown on (Figure 2.13). The agents are programmed using the NetLogo environment and their behaviour is derived from "common trends identified by architects and urban planners", thus providing a model for population growth and land use. The available land use categories are residential, commercial, industrial, roads and park.

Figure 2.13: Procedural generation of land use patterns [Lechner et al. 2006]

Further approaches at differentiating between districts in a city are a model creating small or large textured boxes depending on city region (but giving no explanation as to why a region is classified as residential) in [Kelly and Mccabe 2007], a layout optimization approach for the buildings in residential zones [Wang et al. 2007] and a shape grammar for the creation of urban open spaces [Ulmer et al. 2007]. All of which are very specific approaches and only [Wang et al. 2007] base the building and street placement on urban planning models.

# Chapter 3

# The Problem of Generating Believable City Layouts

This chapter discusses the focus of this thesis, how the topic was chosen as a result of the insights gained from the study of related work, and the benefits that can be obtained from pursuing this research.

## 3.1 Analysis of Previous Approaches

Procedural modelling of cities contains several areas of research, as introduced in Chapter 2. In this section we examine where improvements can be made to further the quality of generated cities.

### 3.1.1 Road Networks

The procedural methods for the generation of road networks (Section 2.1) are already quite advanced. Improvements could be obtained by taking into account a larger number of factors apart from population density and terrain. Examples include the age and continent of a city (resulting in old crooked streets in an old city centre and more grid-like patterns in newer areas and also distinguishing between common European and US-American layouts), or the need for fast connections in commercial areas and more quiet in residential ones. Several scientific theories explain road development. They could be incorporated in the road network generation process, see for example [Masuo 1999, Levinson and Zhang 2006, Xie and Levinson 2005, Lammer et al. 2006, Jiang 2007, Scaparra and Church 2005, Coombe 1996, Ralston and Barber 1982].

### 3.1.2 Buildings

Approaches for procedurally generating buildings (Section 2.2) have reached very high standards. Highly realistic cities can be generated using software such as CityEngine [Procedural Inc. 2008a] if enough data is provided in the form of population density maps, height maps, or city plans such as in the case of the RomeReborn project [Procedural Inc. 2008d] and a recreation of ancient Pompeii [Procedural Inc. 2008c]. The software also offers a wide range of different visual styles for designers who wish to create an urban landscape according to their artistic vision, such as futuristic houses of strange shapes [Procedural Inc. 2008b].

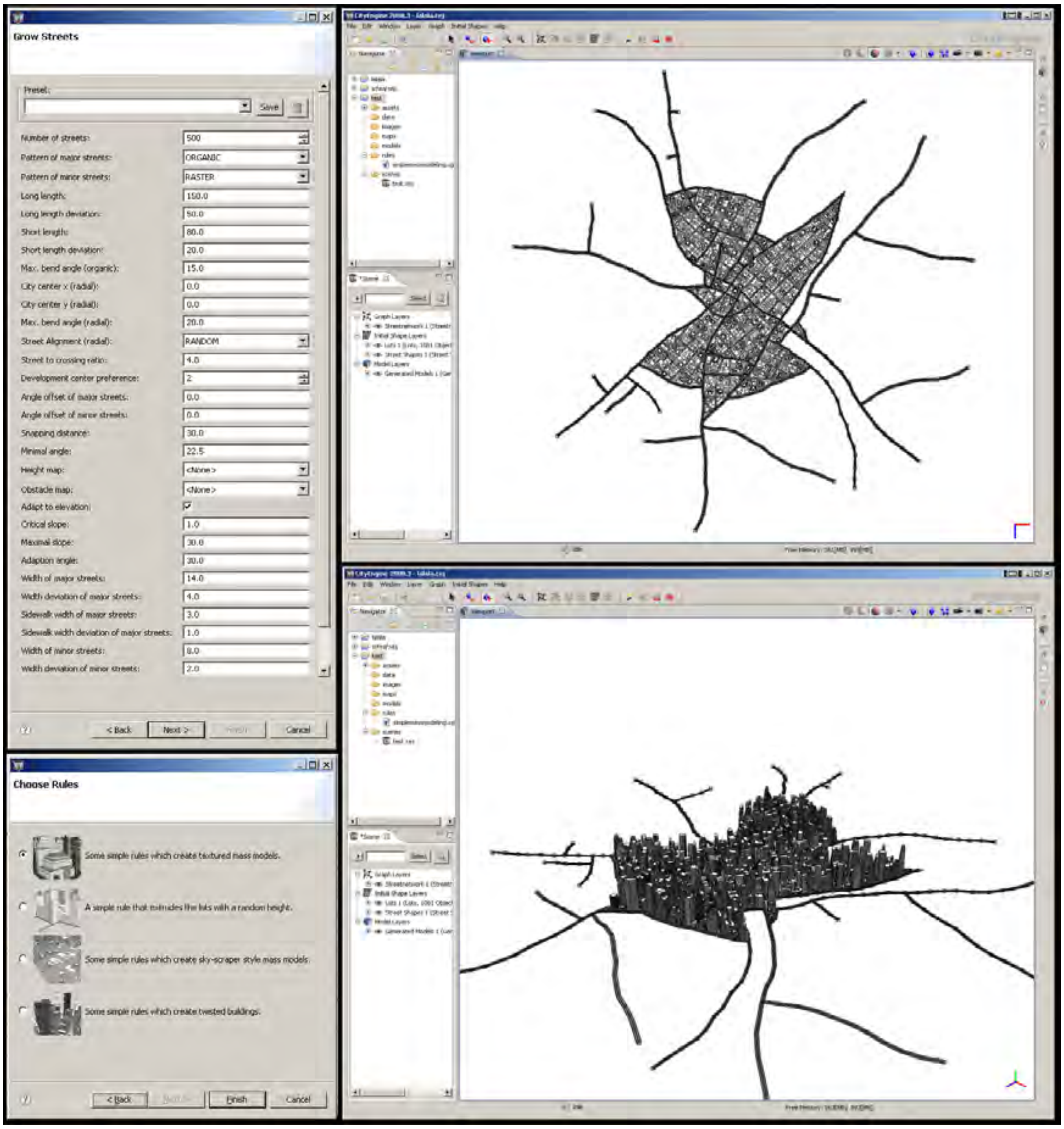
### 3.1.3 CityEngine

CityEngine is by far the most advanced software for the procedural generation of cities available today. What the software lacks, however, is the ability to create believable output without the use of external input. Land use models which would predict the location of different areas in a city, such as residential or industrial districts, are missing from the procedural generation algorithms, leaving the software dependent on knowledgeable users who feed it with the right data. Without external input, CityEngine can only create aesthetically pleasing models of cities whose superficially realistic look does not stand up to closer scrutiny.

Figure 3.1 shows the generation of a city in the CityEngine using the city generation wizard with no external data input. There is a wide selection of input parameters for the generation of the street network. The user can choose from different street patterns (radial, organic and raster) and make many adjustments regarding the geometry of the streets such as width, angle and distance (Figure 3.1 top left). After street generation, the user can choose from 4 different building generation rule sets, ranging from lot extrusion to twisted and textured buildings (Figure 3.1 bottom left). The result (Figure 3.1 top right and bottom right) will always be a street network filled with skyscrapers and the city has no dedicated land use and no believable structure.

### 3.1.4 Land Use

The land use in cities can be modelled using agents (Section 2.3). This approach seems to work well for the four land use categories (residential, commercial, industrial, and park) it covers. The main drawback in this approach is the generation time. A survey of procedural modelling techniques for city generation [Kelly and McCabe 2006] states that, using the agents described, “a city of only limited scale similar to a village can



Number of streets: 500
Pattern of major streets: ORGANIC
Pattern of minor streets: RASTER
Long length: 150.0
Long length deviation: 50.0
Short length: 80.0
Short length deviation: 20.0
Max. bend angle (organic): 15.0
City center x (radial): 0.0
City center y (radial): 0.0
Max. bend angle (radial): 20.0
Street Alignment (radial): RANDOM
Street to crossing ratio: 4.0
Development center preference: 2
Angle offset of major streets: 0.0
Angle offset of minor streets: 0.0
Snapping distance: 30.0
Minimal angle: 22.5
Height map: <None>
Obstacle map: <None>
Adapt to elevation: ☑
Critical slope: 1.0
Maximal slope: 30.0
Adaption angle: 30.0
Width of major streets: 14.0
Width deviation of major streets: 4.0
Sidewalk width of major streets: 3.0
Sidewalk width deviation of major streets: 1.0
Width of minor streets: 8.0
Width deviation of minor streets: 2.0

● Some simple rules which create textured mass models.
○ A simple rule that extrudes the lots with a random height.
○ Some simple rules which create sky-scraper style mass models.
○ Some simple rules which create twisted buildings.

FIGURE 3.1: Generation of a city in the CityEngine software.

be generated over a period of approximately 15 minutes not including the generation of any building geometry or textures". Figure 3.2 shows the generation of such an agent-generated village. This makes land use modelling using agents unsuitable for applications requiring fast generation times. The approach is also limited to only four different types of land use and does not subdivide them further. A desirably realistic land use model should differentiate between, for example, the very different types of residential land use found in a low-class residential area full of high-rise buildings with very high population density, and a high-class residential area with spacious villa estates.

FIGURE 3.2: A 15 minute city generation sequence. Yellow is residential, red is commercial, blue is industrial. Roads are grey. [Kelly and McCabe 2006] after [Lechner et al. 2003]

## 3.2 Aim of this Work

From the analysis of related work it becomes clear that procedural city modelling as a field of research has seen vast improvements over the last 10 years, culminating in the release of a commercially available software [Procedural Inc. 2008a] that, given enough input data, can create believable cities spanning from ancient civilizations to imagined futurescapes, including finely detailed buildings and realistic street layouts. The focus of this thesis is therefore not the improvement of techniques that have already been refined over a decade, but to target a problem that the existing work does not address.

The aim of this work is allowing for *high speed* procedural generation of *structurally plausible* city layouts without the need for *additional data input* by the user.

We define “structurally plausible cities” as cities that exhibit a structure that can be found in existing cities. A simple intuitive example would be a city that has a downtown area full of skyscrapers and a surrounding ring of smaller residences and industrial buildings, whereas a structurally implausible city could have single family dwellings in its center which are surrounded by a ring of skyscrapers interspersed with factories. The research field of urban geography has drawn up many models to describe the spatial structure of cities, and a generated city obeying the rules of a land use model would thus be deemed structurally plausible. Urban land use models are described in Chapter 4.

“Additional data input” is exemplified by such information as population density maps, image maps describing the city structure or satellite imagery, in summary everything that requires extensive knowledge on the side of the user regarding the specifics of city structure. We want to provide a simplistic user interface that lets a non-expert user choose the type of city she wants to create from a few, intuitive parameters, requiring no knowledge of city planning or urban geography.

By “high speed” we refer to generation times in the area of up to one minute, which we deem acceptable for the fast generation of multiple city layouts. This is an improvement of over one order of magnitude relative to the current layout generation approaches using agents which take at least 15 minutes.

What is *not* the aim of this work is to show the evolution of a city over time, e.g. starting with a medieval town and tracing its changes up to present times. We solely want to provide city layouts such as might be found at the present time.

## 3.3 Possible Scenarios of Use

Training and simulation applications in virtual worlds require significant amounts of urban environments. Figure 3.3 shows an example of a modern training simulation in urban terrain, which uses “large and increasingly sophisticated environments” [Pasian 2009]. The models for such simulations are usually modelled by hand, sometimes after real cities. Procedural generation would significantly increase the amount of available training scenarios, offering new and unexpected situations to the trainees and thus avoid distortion of training results through familiarity with the environment. A procedural generation software would moreover give control over the generation of training scenarios directly to the experienced training instructor without the need to employ a modeller unfamiliar with the requirements. By integrating urban land use models into the generation process, a non-expert in urban geography is able to generate virtual cities with an inherently believable structure, something that cannot be accomplished by other software available up to now. A high generation speed would allow the instructor to generate multiple scenarios almost instantaneously and provide him with a choice of several virtual city models. Given the random factors inherent in procedural generation, we believe that it is important to be able to come up with multiple alternatives within a short period of time, so the user can choose the most suitable version among the generated results.

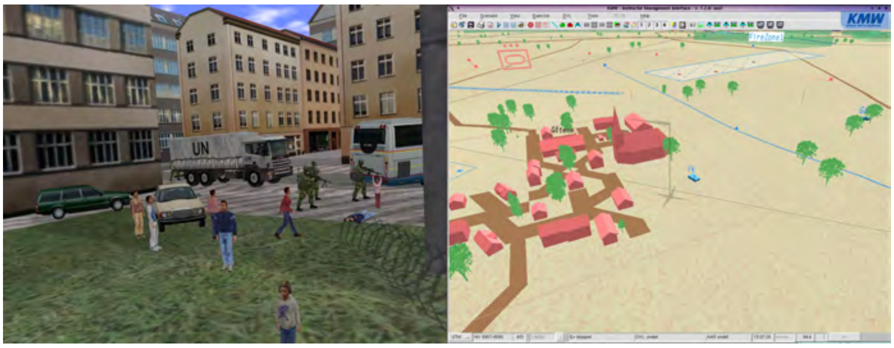


FIGURE 3.3: An urban training simulation called “Military Operations in Urbanized Terrain (MOUT)” [Pasian 2009]. Left: 3D view of the world. Right: instructor interface.

# Chapter 4

# Models for Urban Land Use

The identification and explanation of spatial patterns in the urban structure forms a major topic in the field of urban geography. In a modern city, different districts are distinguishable by their social characteristics (e.g. industry, commerce, high-class and low-class residential areas, etc.). The distribution of those districts within a city is described by a number of principles of urban land use. Simple generic models divide a city into rings or wedges, while more advanced models take into account different local and historic developments.

## 4.1 Simple Models

Urban geography offers three simple models for urban land use. These models describe the placement of different sectors of a city such as business and residential zones.

### 4.1.1 The Chicago School Concentric Model

Developed in 1925 by R. Park, E. Burgess and R. McKenzie [Burgess 1925], this urban land use model was developed after a study of a number of American cities, especially Chicago. It states that a city grows in concentric rings around a central business district (CBD) and suggests that the socio-economic status of households increases proportionally to their distance from the CBD, that is, an industrial zone with low-income, high-crime (”Zone of Transition”) surrounds the business district, followed by working-class, middle-class and finally upper-class residential areas. In this way, good quality housing comes at the cost of long commuting times. The concentric circles expand as the city grows, keeping the overall structure independent of city size. The model is depicted in Figure 4.1.

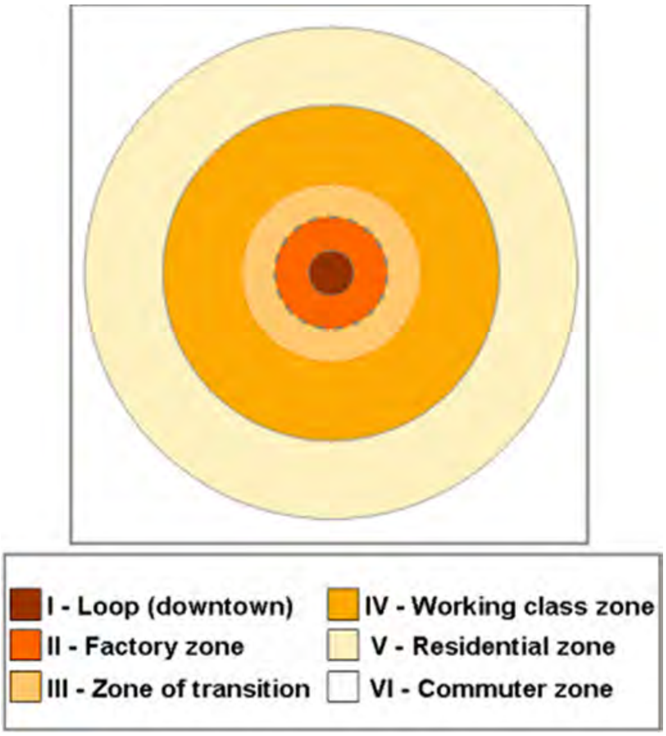


FIGURE 4.1: The Concentric Zone Urban Land Use Model. After [Carter 1981], page 171.

### 4.1.2 The Sector Model

In 1939 H. Hoyt devised an urban land use model based on the study of patterns in American cities between 1878 and 1928 [Hoyt 1939]. Hoyt's model can be seen as an expansion of the concentric ring model (as described in Section 4.1.1). Like the concentric model, it starts with a central business district in the center and includes some concentric processes, but differs in that the surrounding districts mostly grow outward as wedges instead of rings. The general structure of a city according to the Sector Model shows an upper-class residential district extending from the CBD, with middle-class districts on either side of it, while the lower-class district is on the opposite side of the city, containing industry surrounded by working-class housing. With urban growth, the districts extend along transportation lines, forming wedges. Akin to the Concentric Model, the high-income classes have a tendency to build at the current edge of the city and sell their older houses, which are bought by the middle-class who in turn sell to the working-class, thus creating some concentric patterns. Hoyt stresses that the model makes a statement on the relative location of the sectors, not so much on their actual size. Figure 4.2 shows a depiction of the model.

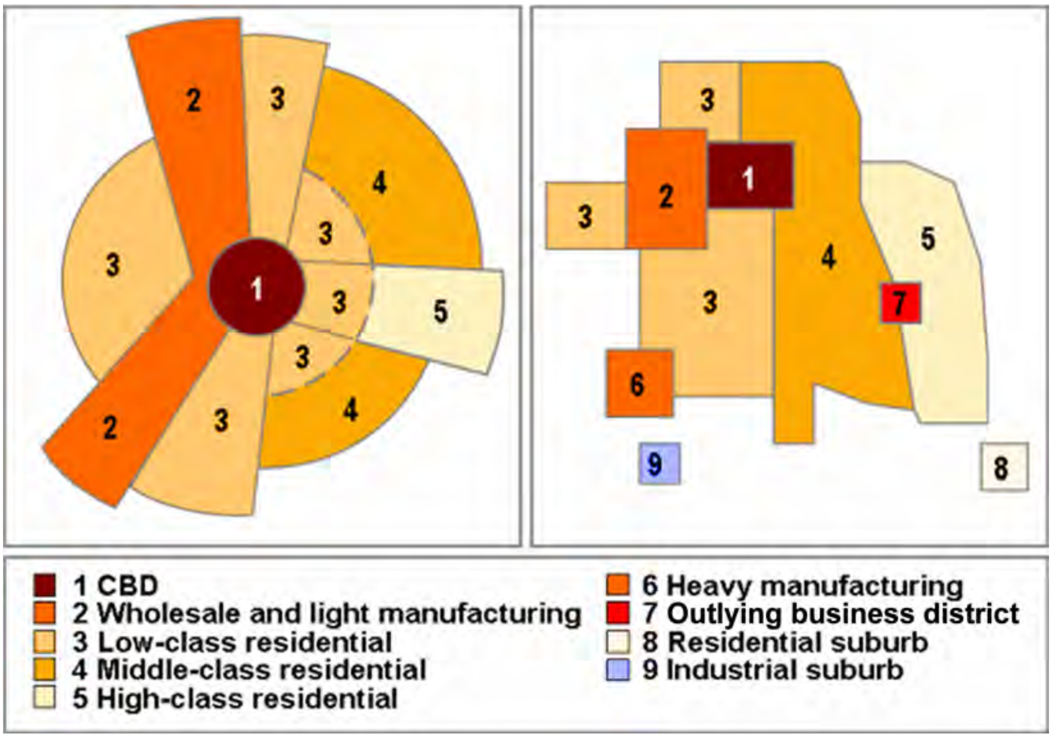


FIGURE 4.2: Left: the Sector Model. Right: the Multiple Nuclei Model. After [Carter 1981], page 171.

### 4.1.3 The Multiple Nuclei Model

The Multiple Nuclei Model was proposed in 1945 by C. Harris and E. Ullman [Harris and Ullman 1945] to incorporate changes occurring in land use with the introduction of cars and subsequent urbanization, which led to more complex cities with multiple centres. The model is based on cities in the USA and north-west Europe. It represents relative locations of major land use categories, stating that the districts are not arranged in a predictable fashion except in relation to the surrounding land use: For instance, commercial and industrial districts beyond the CBD might develop around a government center, university, transit stop or highway intersection. Office and retail centres attract middle-class residential areas, while industrial centres attract working-class residential zones. Figure 4.2 shows an example of the model.

### 4.1.4 Use and Limitations

All three models can be applied; however, which model applies best depends on the age, size and location of a city. The concentric model best describes older and smaller cities, the sector model can be seen as a further development as the city grows, finally leading to a metropolis with multiple nuclei. The models were developed to describe cities in

North America (and in one case also north-west Europe) and do not necessarily apply to cities elsewhere in the world.

However, as [Waugh 2005] states: “It must be remembered that each model will have its limitations. If you make a study of your local town or city, you must avoid the temptation of saying that it fits one of the models - at best it will show characteristics of one or possibly two. Each city is unique and will have its own structure - a pattern not necessarily derived according to any existing model.”

## 4.2 Advanced Models

More elaborate models have been developed which take into account the differing spatial structures of cities in diverse areas of the world. Most of these focus on one of two locations: the cities in Western Europe [Benevolo 1995, Burtenshaw et al. 1991, White 1984] and the cities in North America [Knox 1994, Clark 1982, Carter 1981]. The differences between those two models are significant - apart from cultural differences, most cities in North America lack the historical complexities of urban social, economic and political relationships that are found in Europe. While some show the traces of their colonial roots, the vast majority were erected within the last 150 years, in a time of modern transportation systems, capitalist economy and egalitarianism. The European cities, on the other hand, have gone through various forms of government and social hierarchy never present in North America, such as feudalism, mercantilism, absolutism or socialism. Autocratic rulers have left their marks on European city structures, while other cities formed their own city states [White 1984]. The North American city structure on the other hand is influenced mostly by the advances in transportation technology.

### 4.2.1 The Western European City

The structure of cities in Western Europe is to a large extent influenced by their history - from Roman street structures to medieval merchant quarters, the traces of the past are highly visible in the Western European cities of today. A number of characteristics can be observed in most European cities:

- Important landmarks are castles, churches, monumental squares and other symbols of non-economic power (in contrast to the skyscrapers of American downtowns).
- Buildings are built upward rather than outward, but there are comparably few skyscrapers due to housing regulations.

- Housing traditions differ in the North (spacious housing) and the South (small houses and streets, but large public areas).
- A social and economic gradient from the center to the suburbs, with the wealthy living in or close to the center.
- There is no anti-urban ethos (unlike in the US and also the United Kingdom) driving people out of the cities and into the countryside.

Figure 4.3 shows a general model of a Western European city. The center of the city is formed by its historic core, which is inhabited by the upper and middle-class, the self-employed and also some older working class artisans. The core is sometimes bordered by old city fortifications such as a wall or a moat. Around the core follows the zone of transmission, sub-standard 19th century buildings which are the legacy of industrial growth and inhabited by student, migrants and the indigenous working class. Old industry and associated housing is found along the waterfront, while newer industry lies in the periphery close to social housing for the industry workers. The area close to environmental attractions such as forests or hills also accommodates high status housing, while the middle-class occupies the areas between high- and working-class housing.

### The Historic Core

The European city characteristically possesses an old city center whose layout predates the 19th century. Historically, the ruling powers had a strong influence on the city structure through the arrangement of formal spaces such as churches, squares or markets, which in turn influenced the residential settlements. These historic cores are sharply delineated from the surrounding, newer areas, and fall into three basic types based on that city's style of ruler ship: mercantile, feudal, or absolutist. This difference in rule manifested itself in the distinct structures that can still be seen today in old-city architecture.

**Mercantile Historic Core** The mercantile city (Figure 4.4) was prevalent in Northern Europe. Guilds grouped together everyone involved in a particular trade, leading to districts dominated by different trades. The predominant merchant's house was used as workshop, store and residence of the merchant and all employees, leading to little or no spatial segregation of different classes and thus a social mixture and no concentric zoning. The Hanseatic cities provide good examples of this type.

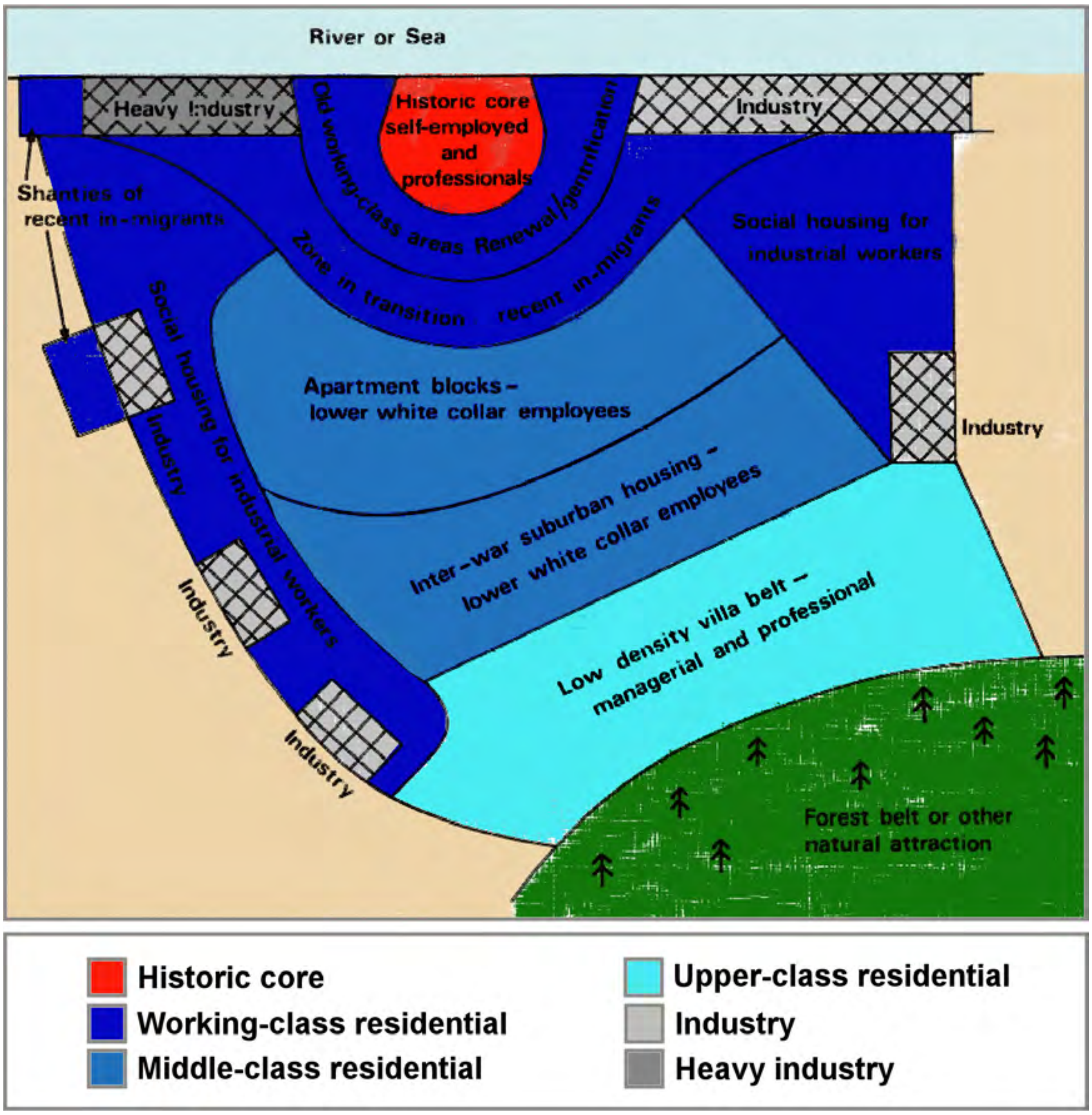


FIGURE 4.3: A Model of the Western European City, after [White 1984], page 188.

FIGURE 4.4: Mercantile historic core. Left: a map of medieval Rostock (source: Stadtarchiv Rostock). Center and right: Historic city center of Rostock today (sources: [Google 2009], [Wikipedia 2009a]).

**Feudal Historic Core** In contrast to the North of Europe, where commercial interests prevailed, Southern European cities were under the feudal rule of several noble families. The often warring factions each built their own family palace, housing a noble family, which would dominate a district. The peasant houses clustered around those palaces, resulting in a spread of high-class areas throughout the city, with lower quality housing in between. This can be seen today in cities such as Siena and San Gimignano in Italy (Figure 4.5).

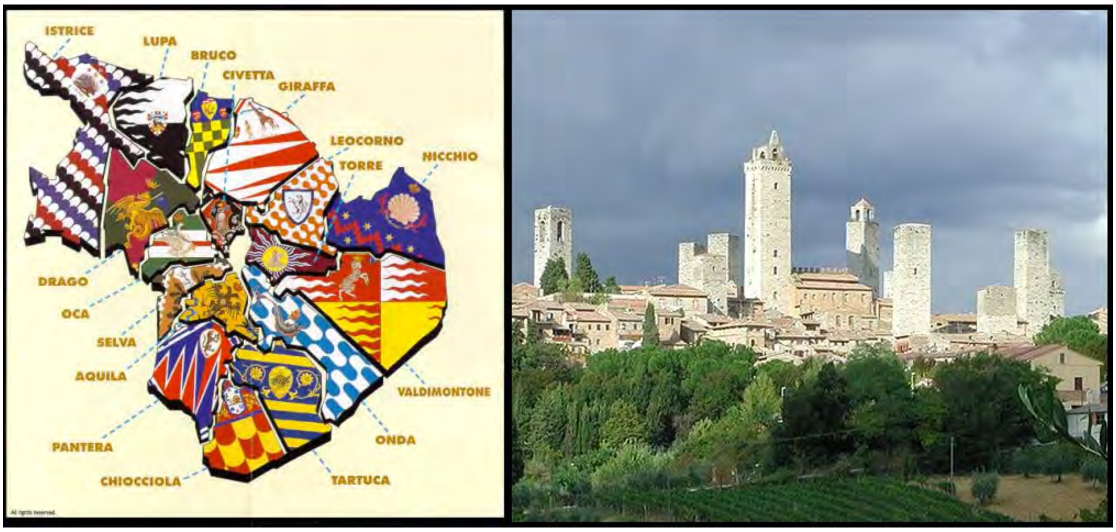


FIGURE 4.5: Feudal historic core. Left: the districts of Siena which compete in the annual horse race, remnants of the city's medieval history [Lufrano 2006]. Right: The towers of the palaces of San Gimignano [Wikipedia 2009b].

**Absolutistic Historic Core** In the subjugated cities, a single family or individual had the power to rebuild large sections of the city following detailed plans. This lead to large-scale geometric structures dominated by the royal palace. The nobles built their houses close to the palace, while the laborers' dwellings were located in the outer areas. Evidence of such structures are nowadays still visible in Karlsruhe, Germany (Figure 4.6) or Stockholm, Sweden.

### City Walls

In some cities, removal of the medieval city walls lead to both a move of the middle-classes into the new open area, as well as the construction of grand boulevards full of representative buildings in a ring around the core. Cologne (Figure 4.7), Bordeaux, Brussels and Sevilla are good examples for this.

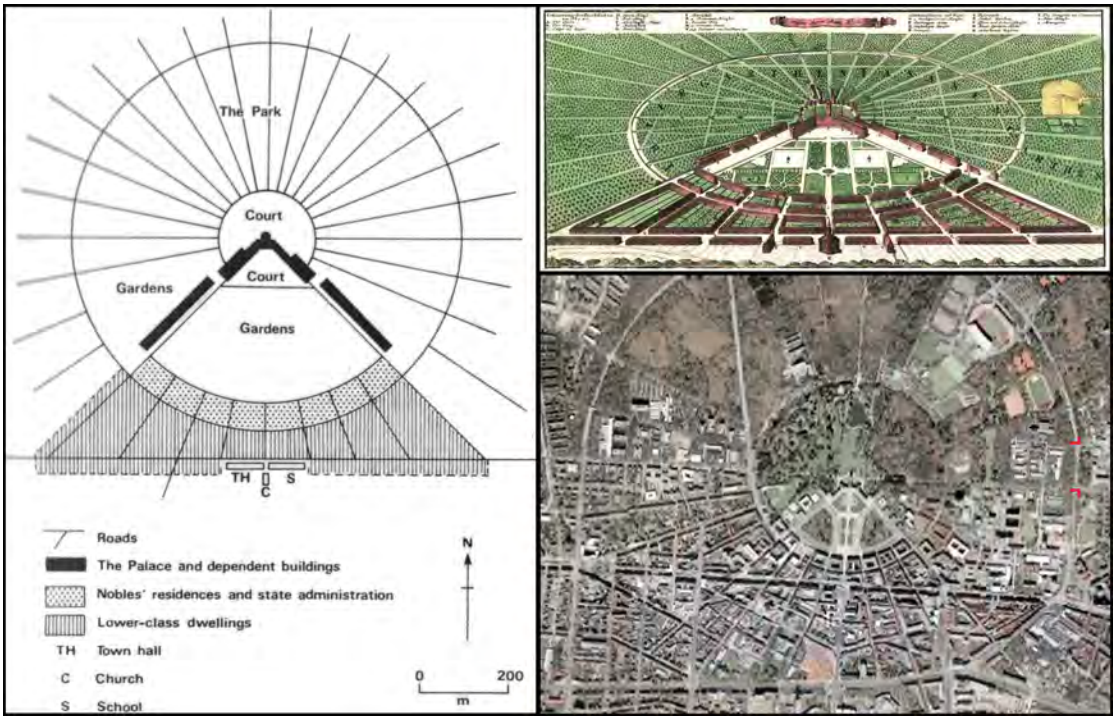


FIGURE 4.6: Absolutistic historic core of Karlsruhe. Left: schematic of 18th century Karlsruhe from [White 1984]. Top right: Karlsruhe in 1721, etching by Heinrich Schwarz. Bottom right: satellite image of Karlsruhe today from [Google 2009].

FIGURE 4.7: City Walls of Cologne. Left: medieval Cologne with city walls [Wikipedia 2009c]. Right: Cologne today with visible city ring in place of the ancient city walls [Google 2009].

#### Social Class and Location

There generally exists a social and economic gradient from the center to the suburbs, with the wealthier population tending towards the city center. High status districts are usually located in or close to the city center, especially in Southern Europe, with some exceptional districts being located in suburbs with attractive surroundings. The lower middle-class can be found especially in suburban areas in post-war public housing, and while some worker districts are in close proximity to the high status districts in the inner city, there is a general tendency for workers towards the periphery.

### 4.2.2 The North American City

In North America, city landmarks are most often skyscrapers and other symbols of economic power. There exists an anti-urban ethos which is manifested in a drift to the countryside of those who can afford the costs such as longer-distance commuting, leading to a social and economic gradient from the suburbs to the center. The layout of major roads has a big influence on the placement of districts, more so than in European cities [Knox 1994, Clark 1982, Carter 1981].

#### The Central Business District

Characteristic for North American cities is the Central Business District (CBD) which makes up the city core. The CBD is the hub of economic, social and political life in the American city. The overall spatial structure of the CBD (Figure 4.8) is dominated by a high density core that contains the retail, office, entertainment and civic zones and a lower density frame with zones of warehousing, hotels, medical and education facilities and manufacturing.

#### Historic Development

The cities in North America have undergone several phases of economic and demographic development, the main factors of which were: “changes in social structure and lifestyles, innovations in building materials and construction techniques, innovations in urban transportation and changes in the legal framework of land ownership, land use law and land use policy” [Knox 1994]. The most important of those factors was the progress of transportation systems, which influenced the spread of goods and people.

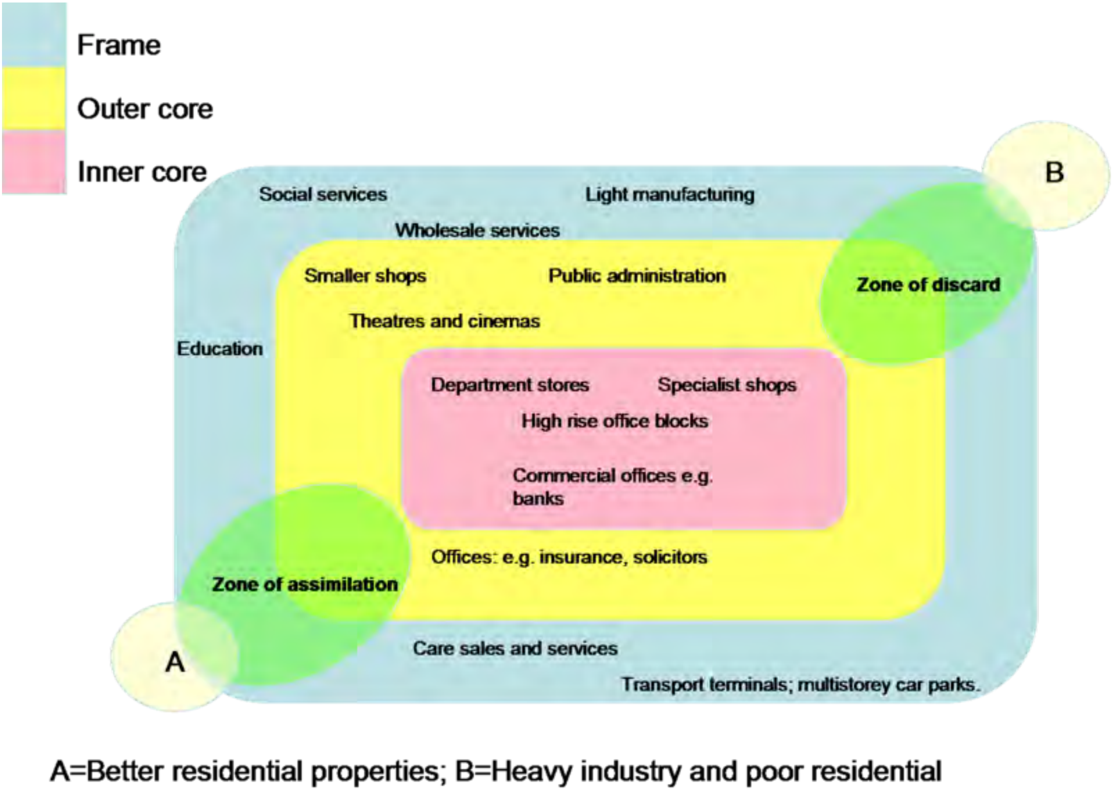


Figure 4.8: Central Business District. Source: [Knights 2009] after [Waugh 2005]

**The Preindustrial City (before 1840)** The city before 1840 was affected by the lack of intra urban transportation, which led to very compact cities with land use patterns similar to those in Europe: the core of the city was home to the elite classes, while the very poor lived on the fringe. The intermediate areas were occupied by a number of socially mixed districts following distinct occupations, reminiscent of the mercantile city of Europe. Everything was within walking distance.

**The Transitional City (1840 - 1875)** With the onset of industrialization, railroads, rapid growth, and the absence of land use laws, the image of the cities changed: the center transformed into an area of specialized industrial and commercial use; residences started to cluster by income instead of occupation; the main means of transport were horse cars and railroads; the development of high-rise building technology led to the first skyscrapers being built in the city center; railroads attracted factories and warehouses, but repelled all but the lowest grade of housing. The rich, as the only ones able to afford the high costs of railroad tickets, moved to the fringes.

**The Industrial City (1875 - 1920)** Social segregation, zoning laws and the prevalence of cars led to yet another city shape. The industrial revolution brought widespread electricity, the technology for steel frame skyscrapers and elevators, and cities grew dramatically larger in population and size. A modern infrastructure developed, including roads, tunnel, bridges, sewers and streetlights. As water was a necessity for the manufacturing of textiles, chemicals and metal, factories were usually the first to acquire available land, making them dominant in the spatial organization of the landscape. Around the factories and between railroads in the least desirable areas lay the cramped dwellings of the factory workers. Decreasing transportation costs left the city core to low-income residences, while the middle-class moved away, making room for nonresidential use. The higher class housing stayed on the fringes of the city while factories, warehouses, offices and other commerce were drawn to the city core. Large factories were situated outside the city, leading to some working-class households on the fringes near those factories. Commerce settled along transportation lines, and behind it came blocks of residential areas. All of these factors led to the development of zoning laws - specifications of the land use allowed in certain parts of the city, which in turn led to specialization and segregation that is still common in US cities today.

### The American City in the 21st Century

The changes during the industrial era developed the structure of the American city of today. Figure 4.9 shows a model of a contemporary city in North America.

An important role is played by the highways, which extend like spokes from the central business district. Around the CBD lie the low income residential districts, while a high-class residential district is found near the fringe and the middle-class residences are found throughout the rest of the city. The socio-economic gradient points therefore into the opposite direction as in the Western European cities. How income affects location can be seen very clearly on Figure 4.10, which shows the median household income in the districts of Charlotte, North Carolina.

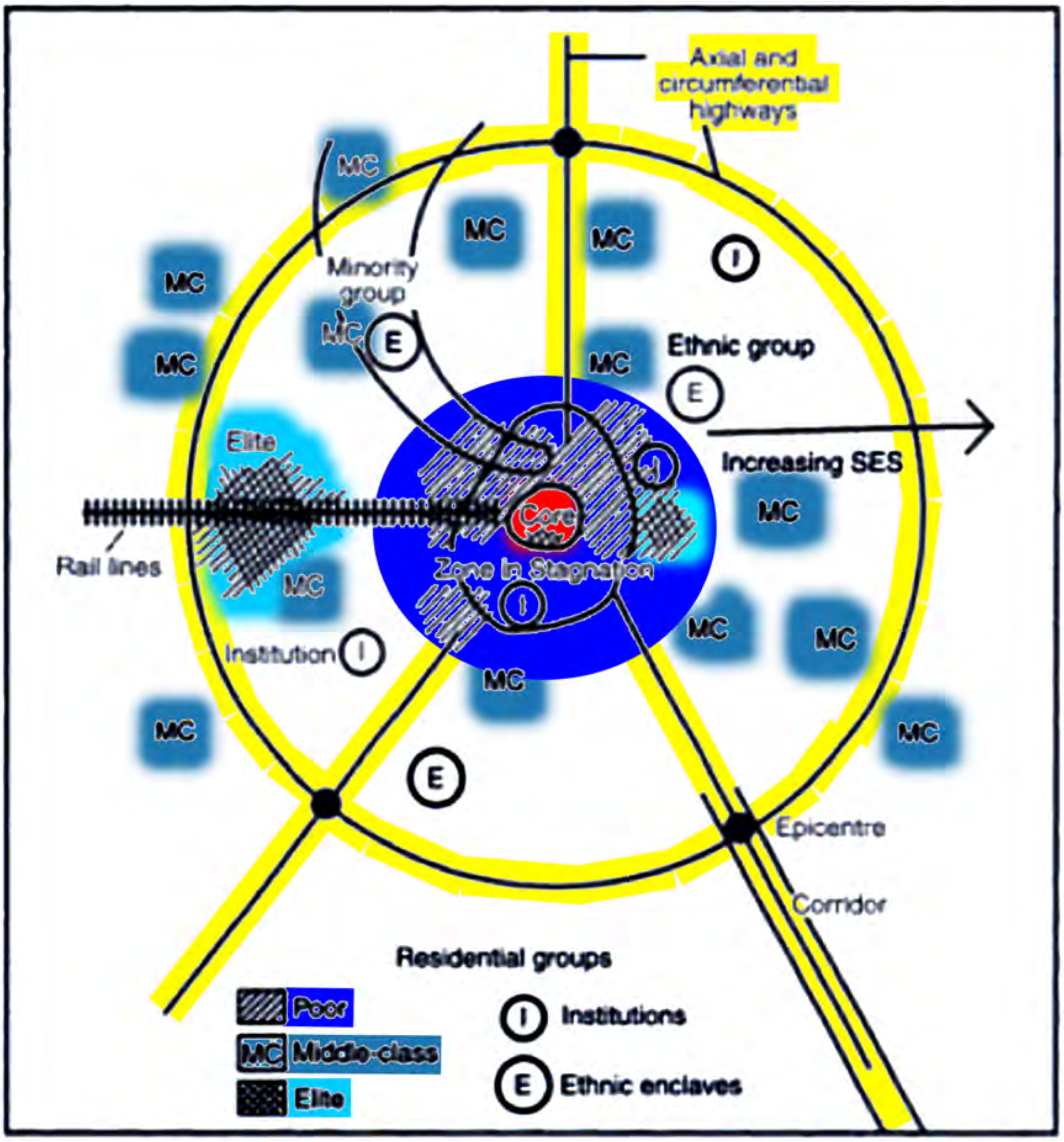


Figure 4.9: White's model of the American city in the 21st century. Source: [Pacione 2007] page 149.

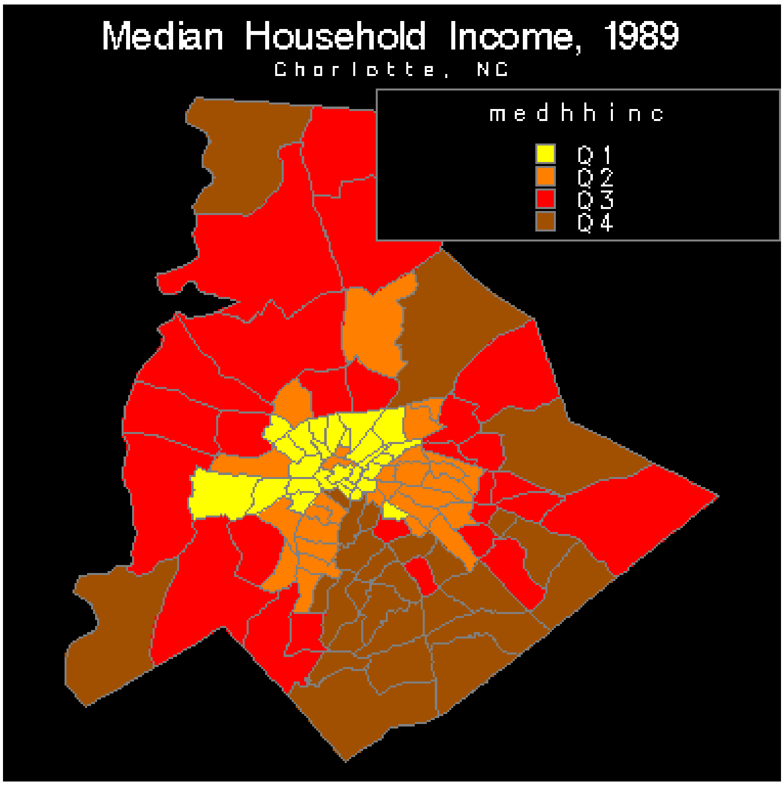


FIGURE 4.10: Median Household Income in Charlotte, NC, 1989 from Q1 (yellow, highest income) to Q4 (brown, lowest income) [Campbell 1998].

## Chapter 5

# Approach and Main Concepts

This chapter describes the concepts we used to solve the problem described in Chapter 3, the procedural generation of structurally plausible city layouts without the need for additional data input by the user at high speed. It gives a short overview over the main ideas, an insight into the development context, and describes how the urban land use models are represented in the algorithm.

## 5.1 Overview over Ideas

The following paragraphs give a short overview over the ideas and subsequent choices we made in solving the problem.

### 5.1.1 Believable City Structure

One main objective of this work is adding a believable structure to automatically generated cities. To achieve this, we turned to the field of urban geography, which offers land use models that describe the spatial layout of a city as described in Chapter 4. These land use models now have to be integrated into a city generation algorithm.

### 5.1.2 Performance

The other objective is a fast generation time. We try to achieve this by avoiding a growth simulation of the city, as this does not benefit our target audience of people who want to generate a 3D city model. Instead we only generate the final state of a city. We also predict that limiting the number of iterations of the algorithm calculating the city layout will speed up generation times, thus we will concentrate on approaches that do not use numerical convergence to find a solution. Numerical optimization over a large number of parameters is very time-consuming.

### 5.1.3 Simplicity of Input

The target audience are people who are untrained in the fields of urban geography, city planning and architecture. The user interface therefore has to provide parameters which are easy to understand, but still offer the user enough control over the generation process to create the cities that he wants. City parameters will not be formulated in terms of street or building geometry, as for example in the CityEngine program, but offer a choice of spatial and historic parameters that can classify different types of cities in a way that is easy to understand by the user.

## 5.2 Context: A Terrain Modelling Framework

The TU Delft and TNO are developing a procedural terrain modelling framework for the automatic generation of virtual worlds [Smelik et al. 2008]. When finished, it will create a 3D terrain model from intuitive user input. “Terrain” in this context means a complete world including earth and water, vegetation, roads and buildings. Figure 5.1 shows the modelling work flow of the terrain framework and the different layers of the terrain map. First, the user roughly sketches a terrain and its global parameters. This input is then used for the procedural generation of a refined terrain, which can be turned into a 3D world using further procedural algorithms. This thesis deals with the realization of the Urban Layer of the framework, as well as to some extent with the Road Layer. The Urban Layer contains all the information regarding cities, such as city structure and dimension as well as land use, which influences which kind of building is later generated at which position in the 3D model. The Road Layer contains the information on roads. The roads form a hierarchy from wide highways over large primary roads to the smallest paths.

The Urban and the Road layer are superimposed on and influenced by the Earth and the Water layers, which contain information about the underlying terrain and the position of rivers and oceans, respectively. The creation of those two basic layers is shown in Figure 5.2. First, the Earth Layer is sketched roughly using a gridded drawing interface. There are 10 different earth terrain types available to the user: Ocean, Grass, Dune, Desert, Hill, Dirt, Low Mountain, Medium Mountain, High Mountain and Alps. The sketching interface works like a drawing program: the user chooses a terrain type and

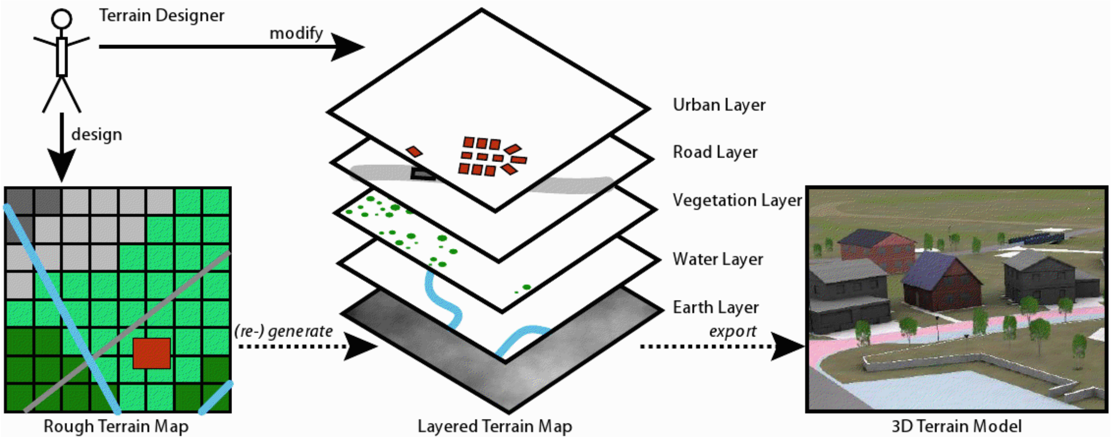


FIGURE 5.1: The Terrain Modelling Framework [Smelik et al. 2008]

draws with it on a canvas, with each terrain type being represented by a different colour. Once the user is satisfied with the sketch, the Earth Layer is generated from the sketch using procedural methods and noise. The Water Layer is generated in the same way from the water terrain. Once the Earth and the Water Layer are generated, the Urban and the Road Layer can be added. The framework also contains a Vegetation Layer for procedurally modelled plants, which at the time of implementation was not yet available in the software so we do not consider it here.

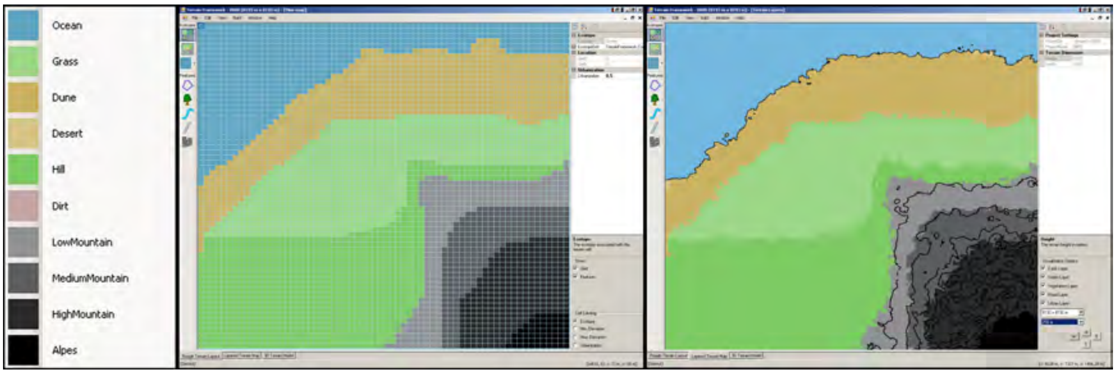


FIGURE 5.2: Earth and Water Layers in the Terrain Framework. Left: available Earth Layer terrain types. Center: sketching interface. Right: generated Earth and Water layers.

## 5.3 Representing Urban Land Use Models

To integrate the urban land use models described in Chapter 4 into the procedural generation of a city layout, we have to find a way to represent them as parameters in the generation algorithm. The following sections describe how this was achieved.

### 5.3.1 Types of Land Use in the City

We define the following types of land use for a city: residential, commercial, industry, and transportation nodes. Each land use type has two or three subtypes.

#### Residential land use

FIGURE 5.3: Residential land use in the UK. Left: low-class. Center: middle-class. Right: high-class. [geographyalltheway 2009]

Residential land use describes land where buildings for human habitation have been erected. It contains three subcategories: High-class, middle-class and low-class residential land. High-class residential areas contain villas and single family houses. Middle-class residential areas contain multi-family houses, and low-class residential areas range from terraced housing to high-rise buildings and trailer parks.

#### Commercial land use

FIGURE 5.4: Commercial land use. Left: hotel [Britton 2001b]. Center: shopping mall [Britton 2001d]. Right: highway commerce [Image Building Systems 2008].

Commercial land comprises office buildings, retail and warehouses. It can be categorized into community-oriented commerce which serves a local area and highway-oriented commerce that settles along major transportation lines.

FIGURE 5.5: Industrial land use. Left: heavy industry by the seaside [Britton 2001e]. Center: heavy industry by the river [Britton 2001f]. Right: light industry [Urban Properties 2001].

### Industrial land use

- Light industry: "may be characterized by the nondurability of manufactured products and a smaller capital investment in plants and equipment, and it may involve nonstandard products, such as customized or craft work. The labour force may be either low skilled, as in textile work and clothing manufacture, food processing, and plastics manufacture, or highly skilled, as in electronics and computer hardware manufacture, precision instrument manufacture, gemstone cutting, and craft work." [Encyclopædia Britannica 2009]
- Heavy industry: "generally requires heavy capital investment in plants and machinery, serves a large and diverse market including other manufacturing industries, has a complex industrial organization and frequently a skilled specialized labour force, and generates a large volume of output. Examples would include petroleum refining, steel and iron manufacturing, motor vehicle and heavy machinery manufacture, cement production, nonferrous metal refining, meat-packing, and hydroelectric power generation." [Encyclopædia Britannica 2009]

### Transportation nodes

Transportation nodes are traffic junctions, train stations, airports, harbours and other areas where large amounts of traffic converge. We differentiate between transport nodes for humans and for goods. [Rodrigue 2006] notes that "population based activities (e.g. residential) are dominant where rail (metro and passenger) and bus infrastructures are converging, while freight based activities (e.g. manufacturing and warehousing) agglomerate nearby high capacity road infrastructures".

FIGURE 5.6: Transportation nodes. Left: train station [Britton 2001a]. Center: freight harbour [Britton 2001g]. Right: airport [Britton 2001c].

### Other types of land use

There are various other types of land use that can be found in a city, such as parks, forests, public spaces, universities, libraries and other public buildings, and many more. We did not include these land uses for several reasons. Most importantly, they are not part of the land use models that we are trying to emulate, nor were they part of other land use models found in the surveyed literature that are not included here, e.g. models of English cities by Robson and Mann, the Dutch cities by Ashworth and Buursink, the German city by Elkins, as well as the European models by Boustedt, Lichtenberger and Neller (see Section B.1). Another reason is that some land uses do not appear as larger units, such as public buildings, but are scattered among other types of land use. We therefore do not model them individually and leave it to the person who generates a 3D model out of our land use layouts to intersperse the relevant buildings. The third reason is that some types of land use are predominantly found in the city core and we only model them there. See Section 5.3.2 for an overview over land use in the core.

### Land use distribution ratios in the city

| | | Sydney | Newport News | Manitowoc |
|---|---|---|---|---|
| Residential | High-class residential | 67% | 71% | 71,4% |
| | Middle-class residential | | | |
| | Low-class residential | | | |
| Commercial | Community-oriented commerce | 8% | 13,4% | 10,2% |
| | Highway-oriented commerce | | | |
| Industry | Light industry | 25% | 15,6% | 18,4% |
| | Heavy industry | | | |

TABLE 5.1: Land use percentage in different cities (adapted to include only modelled land use). Sources: [City of Manitowoc 1999, City of Sydney 2005, City of Newport News 2008]

There are no rules included in land use models as to the percentage at which a land use is present in a city. It general we observe that the residential land use predominates. Table 5.1 shows an overview over available data collected on land use in different cities: Sydney, Australia (which, although neither American nor European, exhibits the traits of a North American city) is a city that spans 25 km$^2$ with a population of 168,682. Newport News, Virginia spans 308.3 km$^2$ with 181,913 inhabitants, and Manitowoc, Wisconsin has an area of 44.5 km$^2$ and a population of 34,053.

We can derive from this data that the residential land uses roughly 70% of a city, the commercial land uses 10% and the industry the final 20%.

### 5.3.2 Types of Land Use in the City Core

The different city cores show types of land use not predominant in the rest of the city. We therefore chose to model several land use types only in the city core. They are listed in the following sections.

#### Land use in the European historic cores

The city centers of the Western European city are described in Section 4.2.1. Land use in those historic districts is influenced to a large extent by their history.

- **The Royal Palace**: The royal palace is found in the absolutist core and constitutes its central element.
- **The Family Palace**: Palaces of noble families are scattered through the feudal core.
- **The Church**: A Cathedral, Church or Monastery. The particularly shape the face of feudal cores.
- **The Market Square**: A public square (e.g. a former fish market). Market squares are a dominant element of the mercantile core.
- **The Guild Quarter**: Most important element of the mercantile core.
- **The Civic Area**: An area mainly filled with public office buildings, like the town hall.

#### Land use in the Central Business District

The Central Business District of the North American city is described and pictured in Section 4.2.2. Its areas are arranged in concentric circles, they are:

- **The Inner Core**
- **The Outer Core**
- **The Fringe**

#### Land use distribution in the core

For the city cores, there was no data or model available to us about the ratios of land use. The data we implemented comes from a superficial observation of some city cores but is by no means accurate or verified in a scientific sense and can therefore only serve as an exemplary implementation. It is, however, very easy for an expert in the field to fill in the correct percentages of land use and with this create accurate land use distributions for the city cores.

The following tables show the distribution of land use within the city cores as implemented in the software. Table 5.2 shows the distribution for the mercantile core, Table 5.3 for the feudal and Table 5.4 for the absolutistic core, all in Western Europe. Table 5.5 is for the central business district of North America.

| **Land use** | **Percentage** |
|---|---|
| Church | 25% |
| Market | 25% |
| High-class residential | 10% |
| Middle-class residential | 10% |
| Low-class residential | 10% |
| Community-oriented commerce | 20% |

Table 5.2: Land use distribution in the mercantile core

### 5.3.3 Influences on the Layout of a City

Analysing the land use models, we can see that the following parameters have an influence on where a certain type of land use can be found in a city: the type of terrain, the area within the city, the proximity to other types of land use as well as the proximity to highways and to bodies of water such as rivers or oceans. The following sections describe how each of those parameters influences the location of different land uses in a city.

| Land use | Percentage |
|---|---|
| Palace | 19% |
| Church | 22% |
| Market | 13% |
| High-class residential | 4% |
| Middle-class residential | 13% |
| Low-class residential | 12% |
| Community-oriented commerce | 12% |
| Civic | 11% |

TABLE 5.3: Land use distribution in the feudal core

| Land use | Percentage |
|---|---|
| RoyalPalace | one, mandatory |
| High-class residential | 30% |
| Middle-class residential | 30% |
| Low-class residential | 40% |

TABLE 5.4: Land use distribution in the absolutistic core

| Land use | Percentage |
|---|---|
| Inner Core | 20% |
| Outer Core | 40% |
| Fringe | 40% |

TABLE 5.5: Land use distribution in the Central Business District

### Terrain type

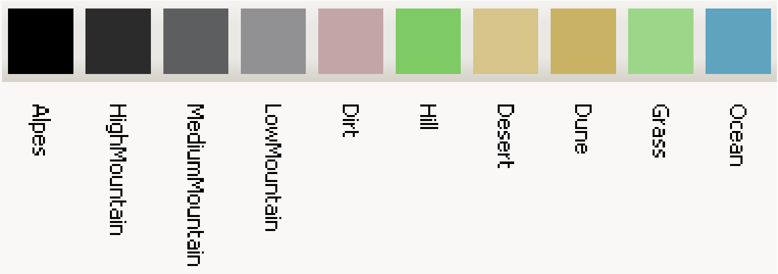


FIGURE 5.7: Terrain types of the Earth Layer

There are 10 different terrain types on which a city can be built, as described in Section 5.2: Ocean, Grass, Dune, Desert, Hill, Dirt, Low Mountain, Medium Mountain, High Mountain and Alps. They are shown in Figure 5.7. Land use, according to the land use models, is to some extent dependent on the underlying terrain:

- The Seaside, represented as Dune terrain, heavily attracts the heavy industry, which needs large amounts of water for its manufacturing processes. The seaside also attracts residential land use, mainly high-class residences who can afford to pay for such a sought-after location.

- The Desert, as well as High Mountain and Alps terrain types, are usually avoided by cities if there is a possibility to build on even land nearby.
- Hills, with their elevated position and nice views, attract mainly High-class residential areas, as well as some Middle-class if there is enough hilly terrain available.

### Area of the city

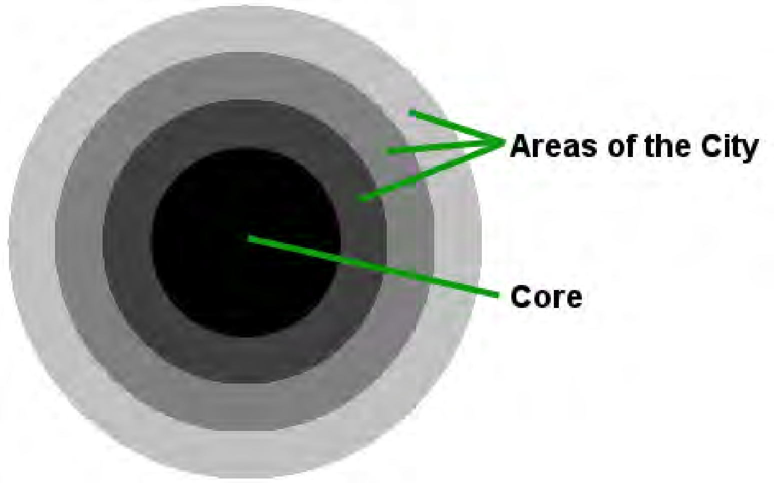


Figure 5.8: Areas of the city

The area of the city plays a large role in the location of different land use. So much, in fact, that the oldest land use models (Section 4.1) based land use predictions entirely on the distance from the city center. We define the areas in a city as concentric rings around the center, as shown in Figure 5.8. Each of the three areas outside the core has a thickness of 1/3 of the thickness of the whole city outside the core. They are used to model the socio-economic gradient contained in the land use models. In the Western European city, the central part of the city will attract high-class residential areas, while the outskirts will accommodate the low-class residences, with the middle-class in between. Inversely, the American city will see the low-class residential areas near the city center and the high-class areas on the fringes of the city. Concentric area rings also play a role in modelling the central business district, whose different types of land use are arranged in concentric circles.

### Proximity to other types of land use

Types of land use can attract or repel each other. Obvious examples would be that nobody likes to live next to a heavy industry area, or that transportation nodes attract commerce. We formulate the following rules:

- Every type of land use attracts other districts of its kind, except for transportation nodes of the same kind. Usually, a residential area is expanded by building more residential areas, industry attracts more industry and so does commerce.
- Transportation nodes of the same kind (i.e. for humans or for goods) are never close. There are rarely two train stations next to each other, for example.
- People strive to better their living conditions and live in as nice an area as possible. As a result, middle-class residential areas are attracted to high-class residential areas, while low-class residential areas are attracted to middle-class residences.
- Light and heavy industry do not “attract” low-income residences, but the areas close to industry are the least favoured areas for settlements, and therefore the residences of low-income workers are often found there through lack of alternative. It also offers industry workers shorter commuting times.

For the city core, we can also determine how different districts will attract or repell each other:

- The royal palace of the absolutistic core attracts high-class residential areas, who in turn attract the middle-class, and the middle-class attract the lower-class residential areas. This is a remnant of the way absolutist rulers structured a city, with the nobles close to the palace and the peasants far away from it.
- The feudal palaces of the feudal core are scattered evenly though the core, never coming close to one another. The reason for this is that they were once the seats of rivaling or even warring families.

### Rivers

Rivers act just like oceans in attracting the industry, mainly the heavy industry.

### Highways

Highways are especially important in North American cities, where they have a major influence on the city layouts. Being major transportation veins, they attract the transportation nodes. Highways also attract commerce, especially the highway-oriented commerce, light industry and to a lesser extent heavy industry. They repel high-class residential areas, because living next to a highway is undesirable, as well as middle-class residences. Low-class residential areas on the other hand can be found close to highways.

### 5.3.4 Modelling Attraction and Repulsion

To model the different influences on the layout of the city described in the previous Section 5.3.3, we assign attraction values for every possible influence to each type of land use. Attraction values are integer values in the range of -999 to 999. A negative attraction value stands for repulsion.

| | |
|---|---:|
| Dune | 100 |
| Ocean | -999 |
| Desert | -999 |
| Hill | -50 |
| Alpes | -999 |
| Dirt | 0 |
| Grass | 0 |
| HighMountain | -100 |
| MediumMountain | -70 |
| LowMountain | -50 |

TABLE 5.6: Attraction towards different terrain types for the Heavy Industry land use type.

Table 5.6 shows an example of attraction values for different terrain types that are assigned to the Heavy Industry land use type. Table 5.7 shows another example of attraction values for other land use types that are assigned to the High-class Residential land use type. For a complete list of all attraction values see Section A.1.

| | |
|---|---:|
| HeavyIndustry | -999 |
| Industry | -500 |
| HighclassResidential | 90 |
| MiddleclassResidential | 50 |
| WorkingclassResidential | -50 |
| TransportationNodeGoods | -100 |
| TransportationNodeHumans | -50 |
| Commercial1 | 0 |
| Commercial2 | 0 |
| Market | 0 |
| Church | 0 |
| Civic | 0 |
| Palace | 100 |
| RoyalPalace | 500 |
| CBDInnerCore | 0 |
| CBDOuterCore | 0 |
| CBDFrame | 0 |

TABLE 5.7: Attraction towards different other land use types for the High-class Residential land use type.

## 5.4 Size and Shape of the City and its Core

The size of the city is initially set by the user as the diameter of a circle. We chose this radial shape as cities who are built on completely even, uniform terrain mostly exhibit circular shapes as they grow evenly into all directions. Examples include Amsterdam and Cologne, which are circular until they reach a natural barrier in the form of a river. See Section B.3 for pictures some more examples of circular cities. This is a simplification, as other cities also exhibit growth e.g. along transportation lines, which result in the growth of finger-like extensions, but we assume this simple case for our implementation. A city boundary also becomes ragged when influenced by external terrain factors such as mountains or oceans, which do not allow the city to grow further into one direction. We model this by classifying terrain types as valid or forbidden terrain, i.e. terrain on which the city can grow and terrain on which nothing will be built. Forbidden terrain includes ocean, desert, high mountains and alps. We realise the forbidden terrain by generating a binary map from the given terrain and limiting the city area to the valid terrain. This is shown in Figure 5.9.

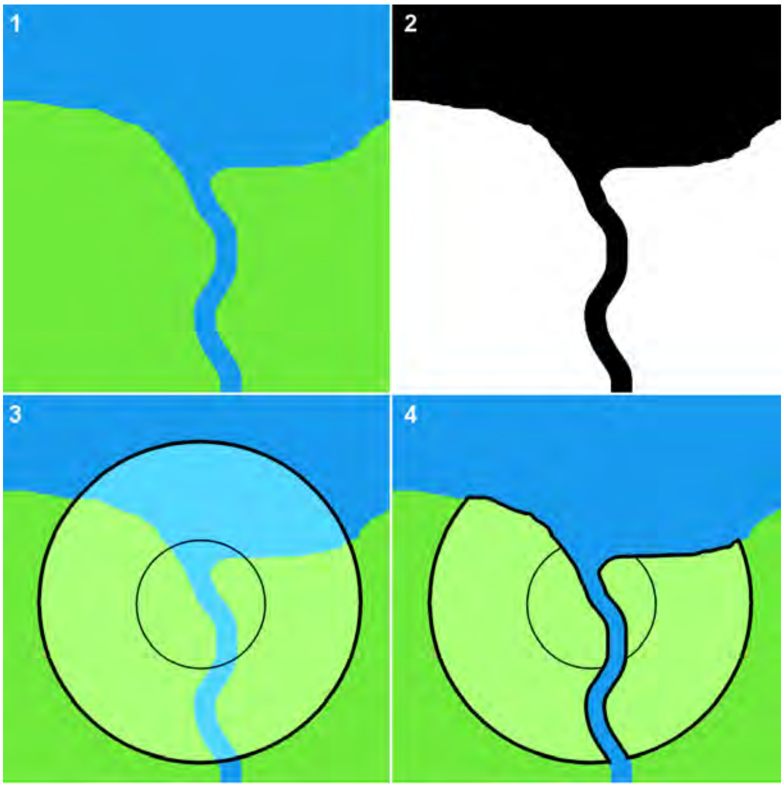


FIGURE 5.9: Influence of ”forbidden terrain” areas on the city shape. 1) Terrain. 2) Binary map of allowed (white) and forbidden (black) terrain. 3) Initial city area. 4) City area after removal of forbidden terrain.

Before implementing the “forbidden terrain” binary map, we used another approach to split a city into different parts in case something (e.g. a river or a mountain range) was dividing it. We implemented two clustering algorithms (K-means [Steinhaus 1956] and DBSCAN [Ester et al. 1996]) which would group the district centres of a city together based on their proximity. This was later abolished as the binary map provides a much easier, faster and more elegant solution to the problem.

The size of the core depends on the size of the city. We have looked at many cities and their cores on GoogleMaps [Google 2009] and come to the following estimates: If the city diameter is larger than 6 kilometres, the core does not extend a diameter of 4 kilometres, making this the largest a city core can become in our implementation. If the city diameter is between 3 and 6 kilometres, the core is set to a diameter of 2 kilometres. If the city is smaller than 3km, the city core diameter is set to a third of the city diameter.

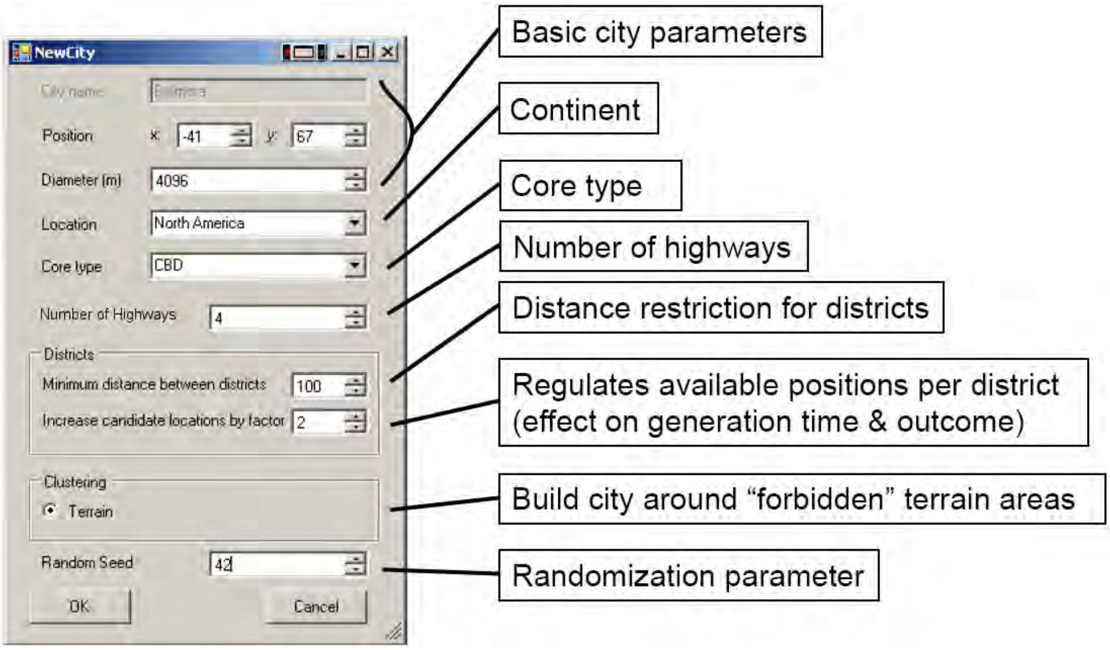
# Chapter 6

# Implementation and Results

This chapter provides the details of the city layout generation algorithm and presents the results of the generation process.

## 6.1 The User Interface

After the Earth and Water layer have been generated as described in Section 5.2, a new city can be created. To start the city creation process, the user selects the corresponding menu item "generate new city" from a context menu in the framework. She is then provided with the city generation interface shown in Figure 6.1.



NewCity

City name [illegible]
Position x: -41 y: 67
Diameter (m) 4096
Location North America
Core type CBD
Number of Highways 4
Districts
Minimum distance between districts 100
Increase candidate locations by factor 2
Clustering
● Terrain
Random Seed 42

FIGURE 6.1: City Creation Menu

The interface allows the setting of several parameters. The parameters are chosen in a way that makes them easy to understand for a non-expert in city design: the user can choose a name and a position for the city on the terrain map, as well as the city diameter. A small diameter leads to the generation of a village while a large diameter results in a metropolis. The user then chooses the continent on which the city is located. The available choices are Western Europe and North America, corresponding to the land use models that are implemented in the system and described in Section 4.2. For the Western European city the user can determine one of three available historic city cores: the mercantile core, the feudal core or the absolutistic core (which are described in Section 4.2.1). A Central Business District (Section 4.2.2) is available for the American city. The next setting is the number of highways in the city. The highways will extend from a ring road around the city center like spokes on a wheel to the outer city limits. The rest of the parameters are more technical and not related to urban land use models. They require that the user become somewhat familiar with the generation process, but once understood they are easily manipulated toward the desired effect. The user can set a minimum distance between district centers to achieve a more even spread and more consistent size of districts in the city. She can also set a parameter that controls the number of locations that a district can assume ("increase candidate locations by factor"), which influences the resulting generated city. The details of this setting are explained in Section 6.2, as a general rule an increase of the parameter leads to bigger clusters of districts of the same kind. The next parameter, Clustering, activates the recognition of terrain areas that cannot be build on, such as ocean, river, or steep mountains, and excludes them from further consideration. The effect is that the city will be built around these terrains and might also be split into several parts (or "clusters") if unfit terrain runs through the middle of the city. The last parameter sets a seed for all randomized processes in the program, which enables the user to replicate a city exactly by setting the same random seed a second time.

## 6.2 Placement of Districts in the City

A city is made up of many districts. In each district, one type of land use is usually predominant. This means there are villa districts, industrial districts, districts with a lot of commerce, and so forth. We model a city layout therefore as an accumulation of districts which are defined by their predominant land use type. The different district types in a city are equal to the land use types described in Section 5.3.1 and Section 5.3.2. Their location will be determined by the factors derived from urban land use models as listed in Section 5.3.3 and their number by the percentages found in Section 5.3.1. This section explains the algorithm behind the automatic generation of a layout through the placement of districts in a city.

Generation of a city layout is a three step process. First, the basic parameters of the city are set by the user through the user interface described in Section 6.1. Then a number of districts are generated with an assigned type of land use but without a location. In the final step, their position within the city is determined.

Through the city creation menu, the user determines the parameters of the city, which are used as the basis for the generation algorithm: location on the terrain, size in diameter, continent, core type, number of highways, minimum distance between district centers and a factor for the number of possible locations per district.

The number of districts is proportional to the diameter of the city. Each district is assigned a type of land use. We know the mean ratios of land use types in a city (Section 5.3.1) and use this as a basis for assigning land use types to districts: Each land use type is assigned a range of numbers between 1 and 100 according to its ratio, which represents the chance of a district being assigned this type of land use. For each district, a random number between 1 and 100 is generated and the land use type belonging to this number is assigned to the district. This is done so that the average land use distribution ratio stays close to the ratio observed in real cities, but the randomness offers some variability each time a new city is generated. With this we want to account for the variations in land use ratios in real cities, which can differ greatly between e.g. industry-dominated cities like Manchester (UK) or the cities of the Ruhr Area (Germany) and mainly residential retirement cities like Sun City (Arizona).

Once all districts of a city are specified, their location within the city is determined. The district placement is influenced by the five different factors stated in Section 5.3.3: type of neighbouring districts (e.g. the high-end residences are unlikely to be situated near an industrial area), terrain type (e.g. industry districts preferring to be near water and high-class residential districts preferring hilly terrain), area of the city (e.g. to represent the social and economic gradient from the center to the suburbs), location of rivers and location of highways.

Following the urban land use models, each district type is assigned different values for each of those factors to signify the degree of attraction or repulsion, as explained in Section 5.3.4. Then, the district placement algorithm assigns the districts to their location.

### 6.2.1 District Placement Algorithm

The district placement algorithm tries to find the ideal location for each district within the perimeter of the city and the city core, respectively. To do so, a number of *candidate locations* are placed within the city limits as a random distribution of points. The city limits were defined by the user through the location and diameter of the city. The number of candidate locations equals the number of district times a factor. This factor is also set by the user though the city creation menu. Once the candidate locations are defined, the districts are placed one by one, until the last district has found its best location.

To calculate the best location for a district $d$, a value is calculated at each candidate location $l$. We call this value *suitability* $S$. Suitability indicates how well a candidate location is suited to accommodate the district. This is also called the “mutual fitness degree” of the candidate location and the district. The calculation of suitability takes into account the city parameters chosen for the urban land use model

For each district $d$ that has to be allocated, we compute the suitability $S$ of each available candidate location $l$, and assign to $d$ the location with the highest $S$. $S$ is a function of five parameters, after the influences on the layout we defined in Section 5.3.3:

- $S_d$: suitability of the location regarding the proximity of the $n$ other already placed districts
- $S_t$: suitability of the location regarding the terrain type
- $S_a$: suitability of the location regarding the area within the city
- $S_r$: suitability of the location regarding the distance from the nearest river
- $S_h$: suitability of the location regarding the distance from the nearest highway

**Placement Relative to Other Already Placed Districts** Equation 6.1 shows the calculation of $S_d$ - the suitability of a candidate location $l$ for district $d$ regarding the proximity of the $n$ surrounding *already placed* districts. $A_{d_i}$ denotes the attraction value towards the land use type of district $d_i$. $\Delta_{d_i}$ is the actual distance to $d_i$, while $\Delta_{min}$ is the minimum possible distance between district centres as set by the user.

The repulsion between two districts becomes infinite in case they are closer together than the minimum district distance, which means that the candidate location is discarded as its suitability is set to negative infinity ($S_d = -\infty$). Otherwise, the attraction is inversely proportional to their distance times the attraction value.

Attraction coefficients are assigned integer values between -999 (high repulsion) and 999 (high attraction), which depend on the chosen city type (e.g. European with mercantile core) as defined in Section 5.3.4. Distances are measured in pixels.

$$S_d = \begin{cases} -\infty & \exists \Delta_{d_i} < \Delta_{min} \\ \frac{\sum_{i=1}^{n} S_{d_i}}{n} = \frac{\sum_{i=1}^{n} A_{d_i} * 1/(\Delta_{d_i} - \Delta_{min})}{n} & \text{otherwise} \end{cases} \tag{6.1}$$

**Other Factors** The suitability regarding the other four factors is taken directly from the attraction values (Equation 6.2, Equation 6.3, Equation 6.4, Equation 6.5).

$$S_t = A_{t_l} \tag{6.2}$$

$$S_a = A_{a_l} \tag{6.3}$$

$$S_r = A_{r_l} \tag{6.4}$$

$$S_h = A_{h_l} \tag{6.5}$$

Listing 6.1 shows the code for the suitability calculation of a candidate location.

```
public double districtsuitability(List<DistrictDefinition> alldistricts, int ↩
  distanceThreshold, int corediameter, double[] weights) {
          this._suitability = 0;

          foreach (DistrictDefinition d in alldistricts) {
               double tmp = this._suitability;
               double dist = calculateDistrictDistance(d);
               if (dist >= distanceThreshold)
                    this._suitability += (weights[0] * (this.↩
  _districtPreferences[d.Id] * (1.0 / (dist - distanceThreshold))));
               else {
                    this._suitability = NEG_INFINITY;
                    break;
               }
          }
          if (alldistricts.Count > 0)
                         this._suitability /= alldistricts.Count;

          this._suitability += (this._terrainPreferences[this._ecotope])* ↩
  weights[1] ;
          this._suitability += (this._areaPreferences[this._area])* weights[2]↩
   ;
          this._suitability += (this._riverPreferences[this._riverdistance]) *↩
   weights[3] ;
          this._suitability += (this._roadPreferences[this._roaddistance])* ↩
  weights[4] ;
```

```
            return this._suitability;
        }
```

Listing 6.1: Calculation of the suitability of a candidate location

The values for each parameter are then weighted ($w$) according to their importance for the city type (e.g. distance from highways plays a greater role in American cities). Table 6.1 shows the different weights according to the significance of the influence for the model.

| | Western Europe | North America |
|---|---|---|
| Distance | 2 | 2 |
| Terrain | 1 | 1 |
| Area | 1 | 1 |
| Rivers | 0.5 | 0.5 |
| Highways | 0 | 1 |

Table 6.1: Weights of the different influences according to the land use model

The final value for $S$ at a candidate location $l$ is then defined as sum of all five values:

$$S = w_d * S_d + w_t * S_t + w_a * S_a + w_r * S_r + w_h * S_h \tag{6.6}$$

The location $l$ with the highest suitability $S$ for a district $d$ is chosen and the district is placed there. This is repeated until all districts have been placed. The procedure is first performed for all the districts of the city core and then for those in the rest of the city. Within both sets, the placement order of the districts is randomized to generate varying output layouts.

## 6.3 District Shape and Size

After the locations of the districts have been determined, the shapes of the districts have to be created. To do this, we generate a Voronoi diagram [de Berg et al. 2000] of the district centres. A Voronoi diagram is a spatial tessellation. It is created when partitioning a plane with $n$ points into convex polygons in such a way that each polygon contains exactly one generating point and every point in a given polygon is closer to its generating point than to any other. Each Voronoi polygon represents a district of the city.

We chose a Voronoi diagram to determine district shapes as the districts in real cities often take a shape very similar to the irregular polygon shape of a Voronoi region. This idea has been used before by [Glass et al. 2006] and [Sun et al. 2002], who observed that realistic looking street networks for cities can be created using Voronoi diagrams. As districts are usually bordered by large streets, we can use the same idea to create the areas within the streets, the districts. Figure 6.2 shows the streets of Den Haag next to a Voronoi diagram of colliding wavefronts to show the striking similarities between the two structures.

FIGURE 6.2: Left: Map of Den Haag [Google 2009]. Right: weighted Voronoi diagram formed by colliding wavefronts [Kelly 2008]

The size of the districts vary due to the nature of the Voronoi diagram, but their number is related to the overall area of the city. The average area that a district occupies in the current implementation is 0.04 km$^2$ (40000 m$^2$), but it can become much smaller or larger depending on how close the neighbouring districts are.

Two diagrams are created, one for the core and one for the rest of the city. They are then cut off at approximately the city and core limits, creating one circular and one ring-shaped diagram. To achieve this, we create two polygons: one from the convex hull of the core district centres and one from the convex hull of the outer city district centres. The polygons are slightly enlarged so that the districts at the edges of the city have their center points towards the middle of the district area. We then intersect the polygons with the Voronoi graphs to achieve the final city shape. The process is shown in Figure 6.3.

Some noise is added to the Voronoi diagram edges to achieve a more realistic look.

After the Voronoi diagram has been created, the streets can be drawn. A highway is located in a ring around the city core, delineating the core from the rest of the city. A number of highways extends from this ring to the outer edge of the city like spokes on a wheel. The number of highways is set by the user in the city generation menu and located so that they divide the outer city into equal pieces. The edges of the Voronoi diagram are transformed into the primary streets of the city. The secondary roads, i.e. streets within the districts, are generated after the algorithm of [Parish and Müller 2001]

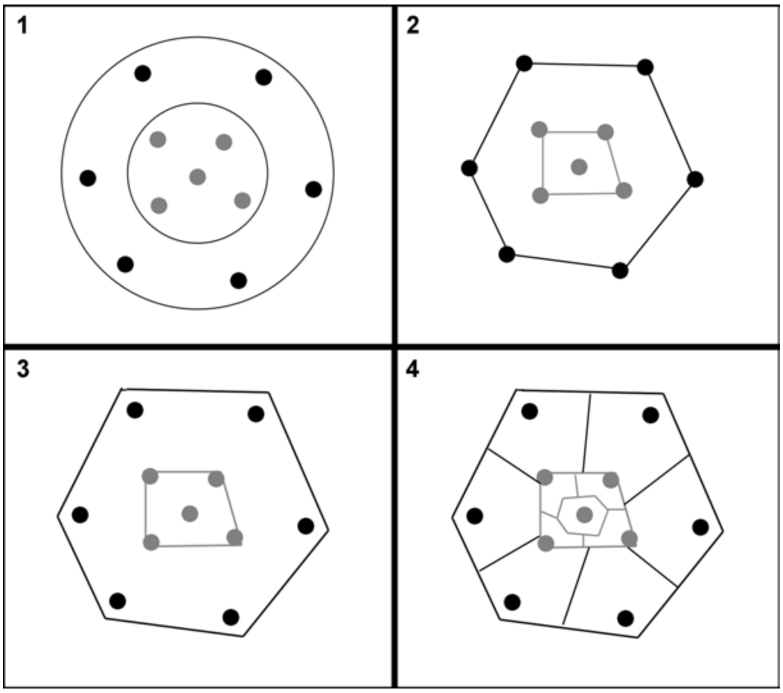


FIGURE 6.3: The final city shape is created: 1) District centres in city limits. 2) Convex hulls of the district centres. 3) Expanded convex hulls. 4) The Voronoi diagrams cut against the convex hull polygons.

that is described in Section 2.1. The implementation of secondary road generation was not part of this work.

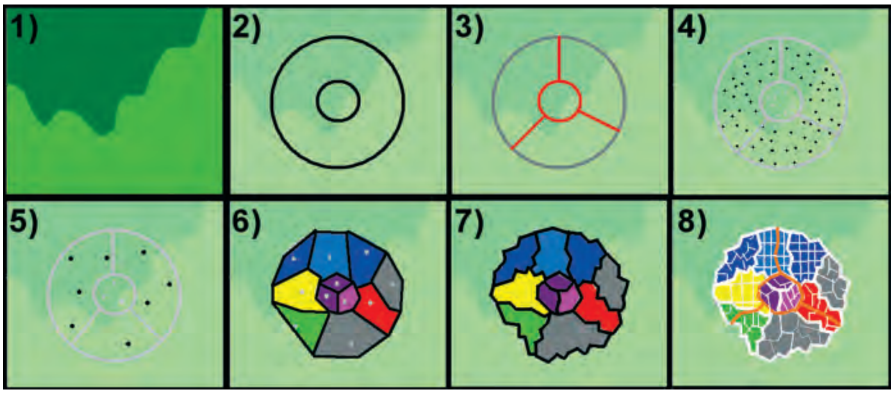


FIGURE 6.4: Creation of a city. 1) Terrain 2) City limits 3) Preliminary highways 4) Candidate locations 5) District locations 6) Voronoi graph 7) Noise 8) Streets

Figure 6.4 sums up the city generation process and shows the progressive placement of districts in the city. First the city size and location on the terrain are defined by the user (2). The (preliminary) highways are drawn automatically around the core and leading out of the city, possibly towards other cities (3). For the placement of the districts, a number of candidate locations are generated based on a random distribution (4) and the best locations chosen for placement of the districts (5). After the locations of the districts have been determined, a Voronoi diagram is generated from the location points, leading to a polygon shape around each location which describes the district limits in such a way as to equally divide the terrain among the districts (6). Some noise is then added to the diagram to achieve a more realistic look (7) and a street network is generated according to known methods [Parish and Müller 2001] (8).

## 6.4 Results

This section shows some generated cities with various parameter variations. The different types of land use are colour-coded as follows (Figure 6.5):

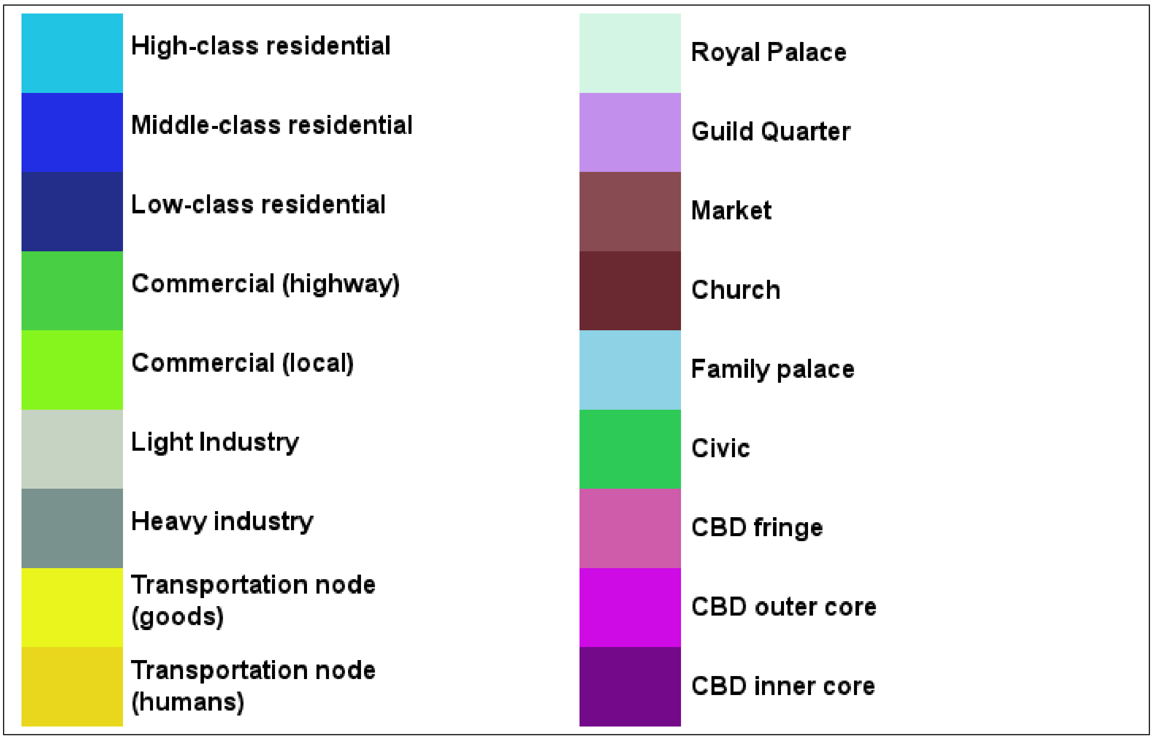


FIGURE 6.5: Colour code for different types of land use in the city (left) and the core (right).

**Terrain** Figure 6.6 demonstrates the effect of the terrain on the city layout. Notice how the industry moves towards the oceanside and how the high-class residential districts prefer the hilly terrain. When a bay is present, the districts are placed around it as a consequence of the "forbidden" ocean terrain.

**Rivers** Figure 6.7 shows how a river influences the city layout. The river cuts the city into three parts. The industry is attracted by the river and changes position to be at the river's edge.

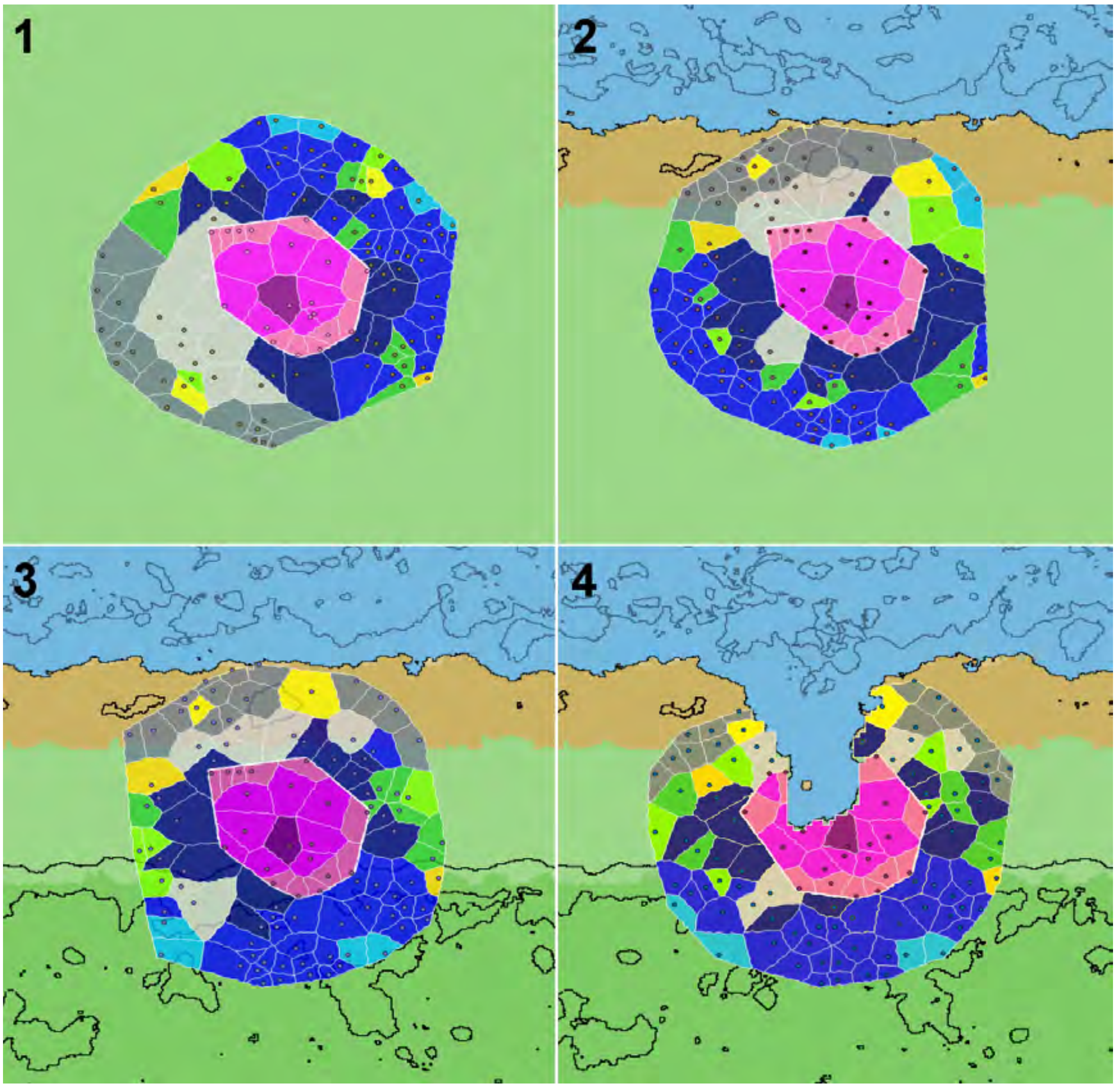


Figure 6.6: North American cities generated with the same random seed under different terrain conditions. 1) On grass. 2) Added ocean and dunes. 3) Added hills. 4) With bay.

**Highways** The district placement in North American cities is influenced by highways. This is shown in Figure 6.8, which shows a city generated with no highways and the same city generated with three highways, once with the highways and roads visible and once with invisible highways to highlight the district placement which is obscured by the roads. Notice how the transportation nodes and the commerce are attracted to the highways, while the high-class residential districts are placed away from the highways.

**Western Europe vs. North America** Figure 6.9 shows four cities with different cores. City 1, 2 and 3 are European with different historic cores. One can clearly see the socio-economic gradient from the central area near the core (high-class residential areas) to the fringe (low-class residential areas)

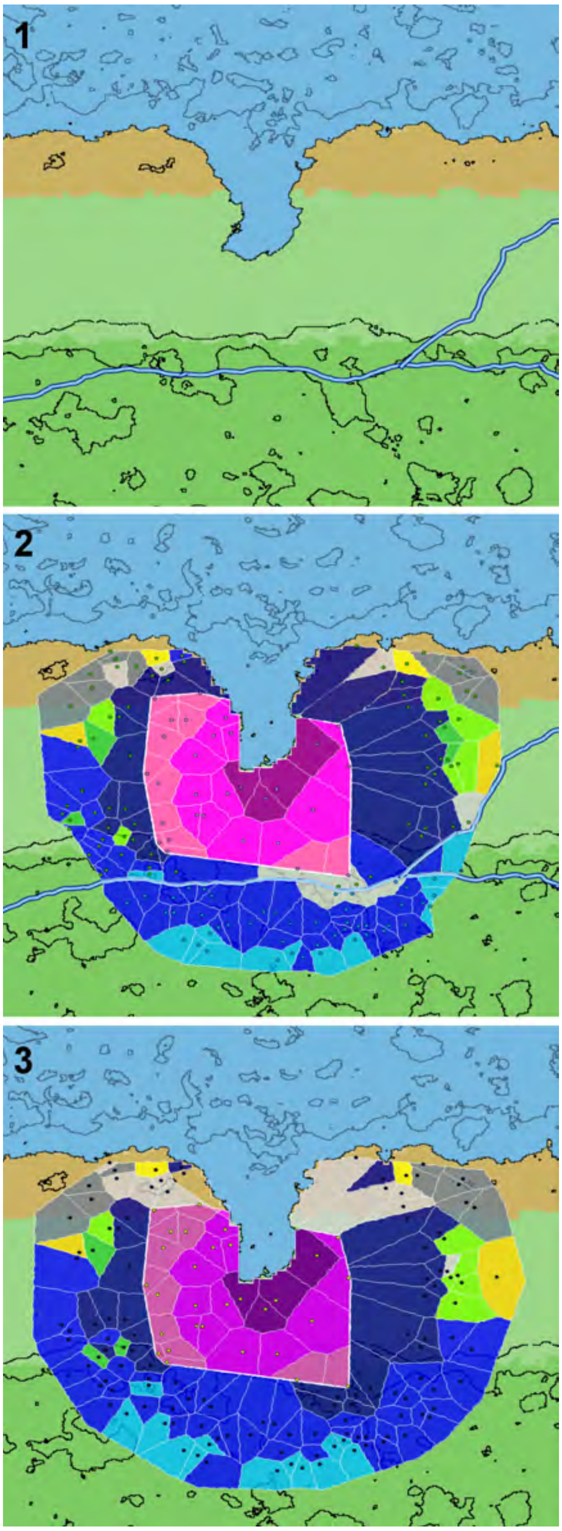


Figure 6.7: The influence of a river on the city layout (American city with CBD and no highways): 1) Terrain with river. 2) City. 3) The same city without a river.

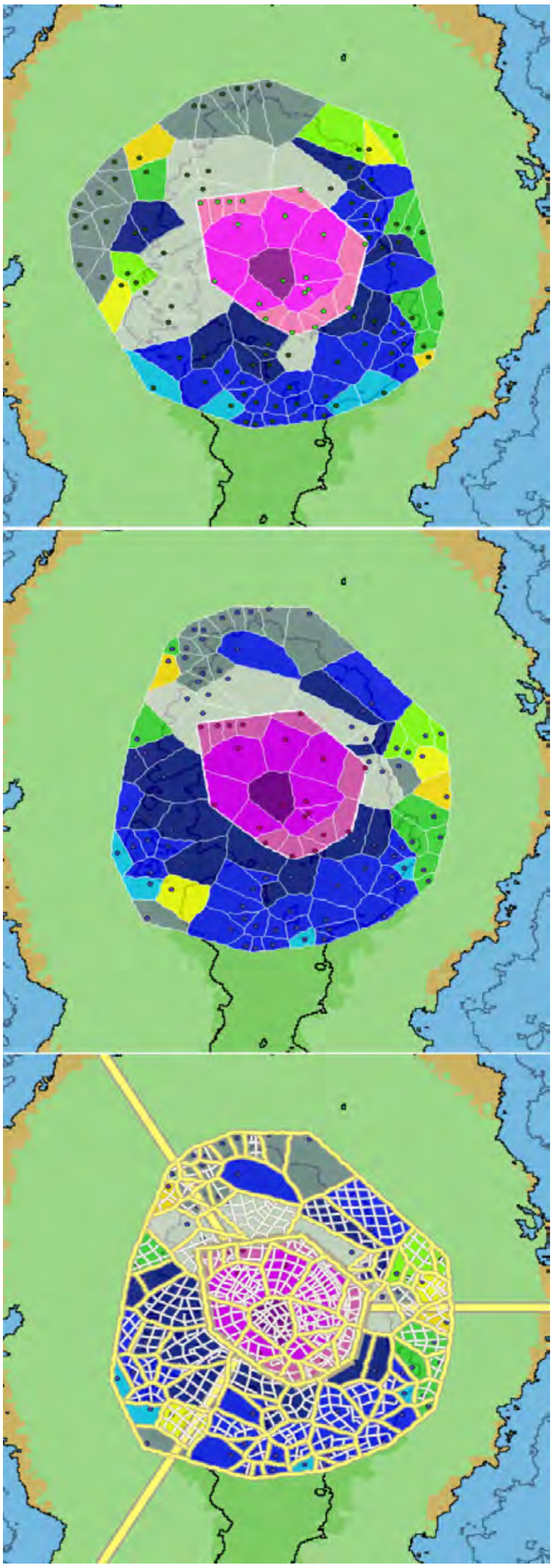


FIGURE 6.8: Influence of highways on the district distribution in the North American city. 1) City without highways. 2) City generated with 3 highways but without highways drawn. 3) Highways and roads drawn over the city.

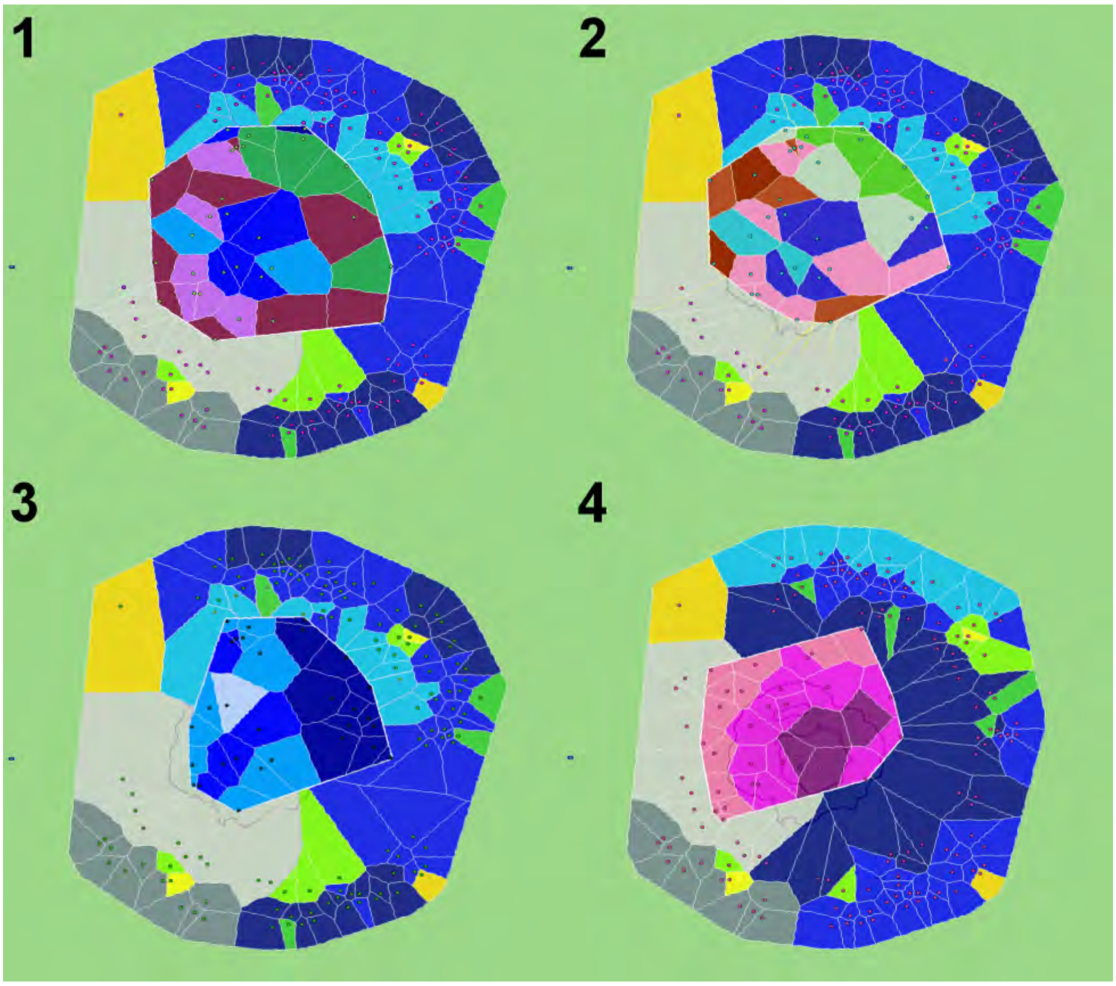


FIGURE 6.9: Different city cores. 1) European city with mercantile core 2) European city with feudal core 3) European city with absolutistic core 4) American city with CBD.

**Influence of the Number of Candidate Locations** Varying the amount of candidate locations available for district placement has an effect on the final distribution of districts, as can be seen on Figure 6.10. The higher the number, the more do the attraction values cause districts to conglomerate if their attraction is high. This effect can be counterbalanced by increasing the minimal distance between district centres.

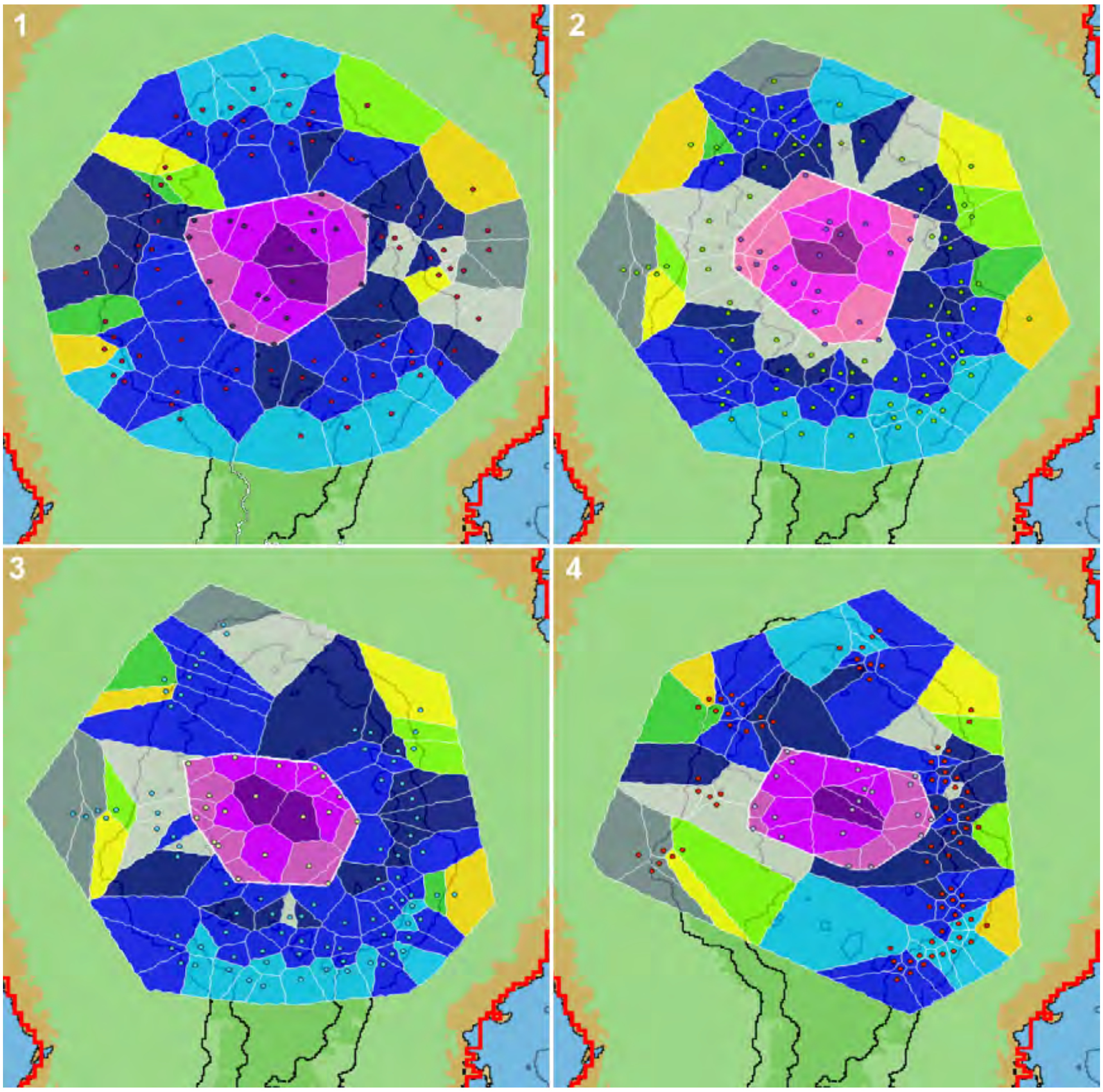


Figure 6.10: Influence on the district distribution by varying the amount of candidate locations. 1) Number of locations = number of districts. 2) Number of locations = 2x number of districts. 3) Number of locations = 3x number of districts. 4) Number of locations = 9x number of districts.

**Influence of the Minimum District Distance** Figure 6.11 shows the effects of varying the minimum district distance. The city is North American with a CBD core. The city diameter is 6096 and the number of candidate locations is twice the number of districts. The picture also shows that districts are left out of the city in case there are no possible locations where they can be placed given the constraints of city area and minimum distance to other district centres.

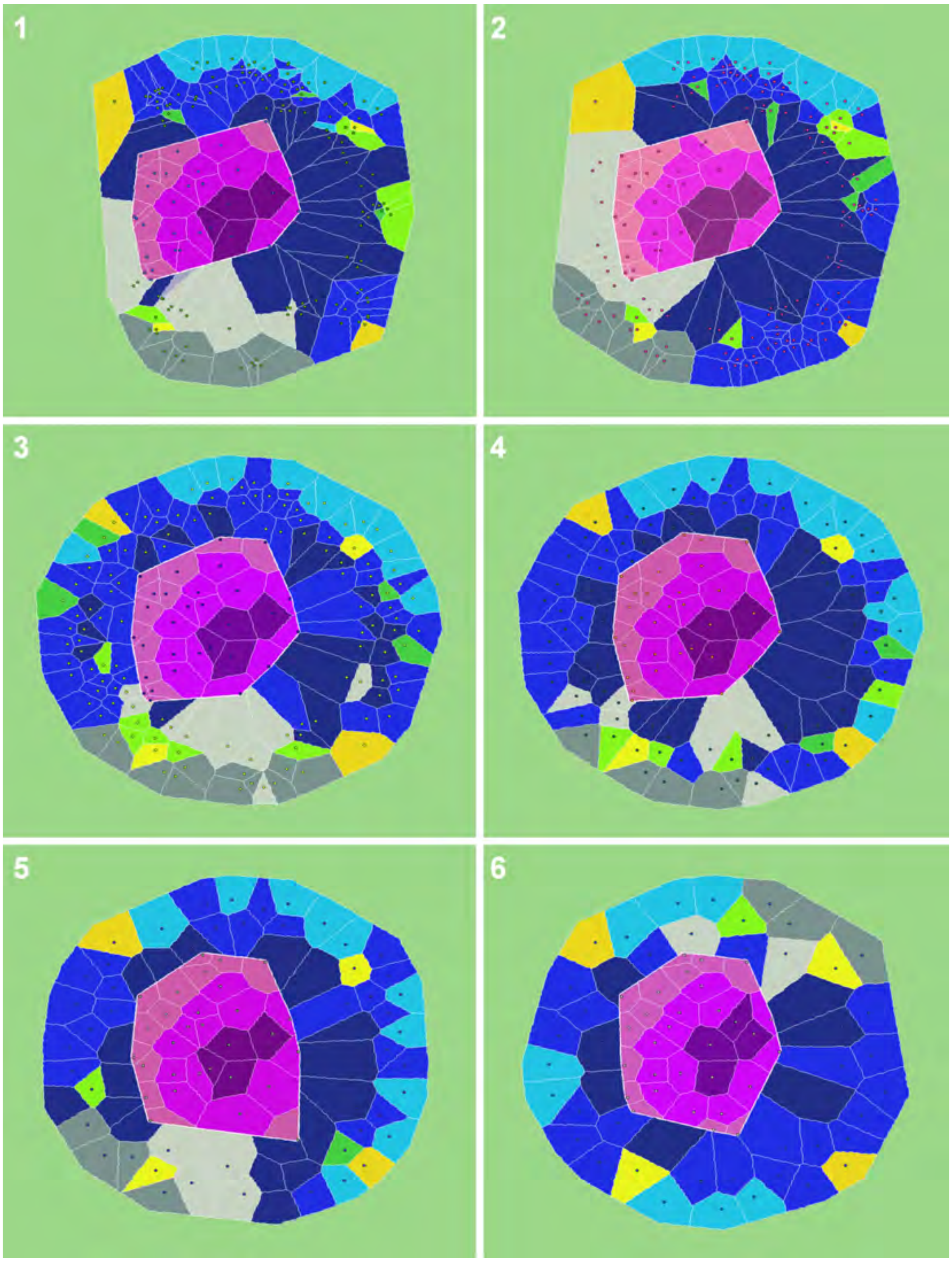


FIGURE 6.11: A city with varying minimum district distance. Minimum distances are: 1) 1. 2) 100. 3) 200. 4) 300. 5) 400. 6) 500.

**Performance** The generation times of cities of varying sizes are shown in Table 6.2 and Figure 6.12.

| **city diameter** | **generation (time in seconds)** |
|---:|---:|
| 1000 | 0.30 |
| 2000 | 0.44 |
| 3000 | 0.67 |
| 4000 | 1.30 |
| 5000 | 1.95 |
| 6000 | 2.64 |
| 7000 | 4.28 |
| 8000 | 5.61 |
| 9000 | 6.27 |
| 10000 | 8.58 |
| 11000 | 15.55 |
| 12000 | 20.02 |
| 13000 | 24.33 |
| 14000 | 25.86 |
| 15000 | 29.58 |
| 20000 | 46.53 |

TABLE 6.2: Performance measurements for the generation of a city including the district placement and the Voronoi diagram generation.

The computer system and programming environment used to develop the implementation were:

- System:
  CPU: Intel Core2 Duo 3.00 GHz
  Memory: 2.00 GB RAM
  Operating System: Microsoft Windows XP Professional SP3
- Programming environment:
  Programming language: C # [Microsoft Corporation 2001]
  Programming environment: Visual Studio .net 2005 [Microsoft Corporation 2005]
  Libraries: QuickGraph [de Halleux 2007] (Voronoi diagrams and roads), OpenSceneGraph 2.7.8 [Kuehne and Martz 2007] (3D terrain), LibCUDA [NVIDIA 2009] (terrain map)

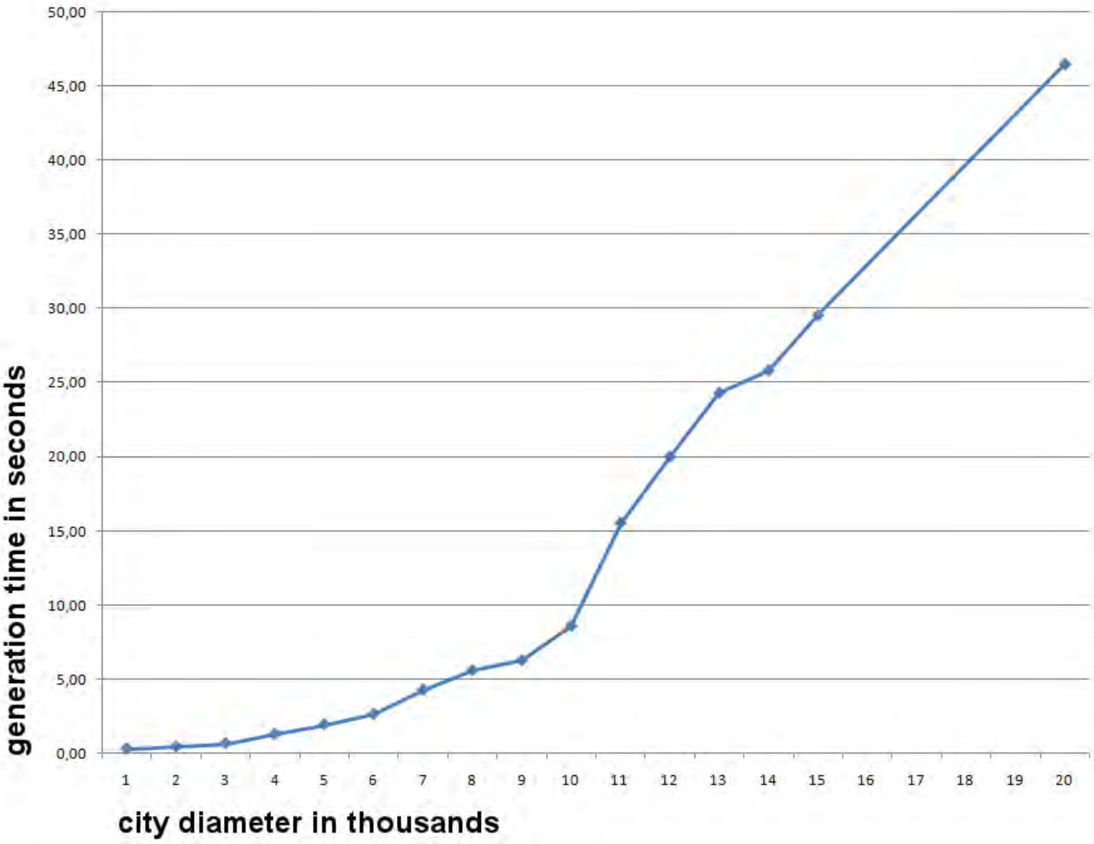


FIGURE 6.12: Performance measurements for the generation of a city including the district placement and the Voronoi diagram generation.

# Chapter 7

# Evaluation

To validate our modelling approach, we have consulted experts on urban design and planning. They approved the city models and parameters we use in our algorithm and confirmed the plausibility of the output layouts. This chapter also highlights the advantages and the limitations of this work.

## 7.1 Evaluation by Experts

After the literature research, a meeting with Megan Ng was arranged. She is a recent graduate of the Faculty of Architecture of Delft University of Technology and was working on a similar subject, which concerned the distribution of populations in a city, and had therefore done research on urban land use as well. She validated the land use models that we planned to implement as common and often used models which would lead to good results if used to generate land use in a city.

After implementation, we showed our program and its result to two PhD students at the Faculty of Architecture of the Delft University of Technology who were recommended because of their experience with similar fields of research: Egbert Stolk, who is an expert in Urbanism, and José Beirão, who studied Urban Design and simulates cities. They offered some consultation and opinions.

José Beirão read and discussed the publication on this work [Groenewegen et al. 2009] and looked at a demo of the finished implementation. His overall opinion was: “good work!”. He related his own experience with trying to use the CityEngine software for generating a realistic city, how this was impossible without additional data and how he consequently dismissed the program as unsuitable to his needs. In his view, “CityEngine is clearly aimed at the film industry", meaning that it is very suitable to create pleasant-looking backdrops for movies, which have no need of a city structure. He stated that a software which could generate land use for cities would also be useful to the needs of urban designers, especially in the first phases of the planning process, to generate a large number of prototypes on which the designers can then base their ideas. José stated that the implementation presented in this thesis produces "plausible results". He marvelled at the seeming emergence of patterns that are also visible in real cities, such as the clustering of districts to form small units of uniform land use, although this is not explicitly coded into the algorithm. He suggested that future work could look at how land use is sometimes mixed in districts, and that it would be nice to find rules that describe which percentage of a district that is mainly used for one type of land use is actually made up of other land use types.

Egbert Stolk also read and discussed the publication on this work [Groenewegen et al. 2009] and looked at a semi-finished version of the implementation. His comments are therefore mainly aimed at the concepts. Asked about the land use models and the different cores used he stated that "the cases you have chosen and the sources you use are very good". Like José Beirão, he also comments on how land use is often mixed in a city, but how land use models neglect this: "We often classify city-districts as if they are mono-functional. In believable cities this is not the case: only functions that cannot be mixed are maintained in mono-functional areas. Inner-city cores are probably the most mixed areas we have, combining working, living, recreation, shopping, etc. Newly build neighborhoods are mostly consisting of housing only (which is also one of the main problems).[..] Limiting the size of a single-use district and cutting it into pieces would improve the 'believability' of your outcome." This is an interesting and important point to consider for future work.

In general, we can observe that all the criteria that are listed in Chapter 3 were met by the generated city layouts. The structure is plausible, the generation speed is very fast and no external data input is needed.

## 7.2 Advantages and Limitations

Our approach generates believable layouts of cities consisting of many different kinds of districts. The generation method is based on models of urban land use, thus requiring neither user knowledge on urban design and planning, nor external input data in order to achieve plausible city layouts. Several intuitive parameters allow choosing among several different types of cities based on their historic background, location, size and shape, covering therefore a large percentage of typical cities in the respective continental settings. Our efficient approach generates layouts in a few seconds, a clear performance distinction from much slower agent-based approaches. Expanding the software to include different models of land use is very easy: all that has to be done is the formulation of the land use models as attraction and repulsion parameters. After entering those into the code, a city conforming to the model will be generated automatically.

There are also some drawbacks to our approach. It is limited to generating the land use models that are coded into the algorithm. As of now, there is no user interface which enables the user to enter own models to expand the system and this has to be done by modifying the code. It would, however, be easy to create an interface that translates user parameters directly into the system. We have avoided this because one of our aims was a program for non-expert users, who will not want to enter their own models. Another drawback is that we have no guarantee that our algorithm produces the best possible output model. All we can say is that the output conforms to the land use models that we are trying to emulate and that experts confirm that the output is plausible. To generate a result where every district is placed at its optimum location, we would need to run a numerical optimization process that evaluates the position of every district in regard to every other district, in addition to the parameters given by the environment (terrain, proximity to rivers, highways, etc.). This is a complex problem over a large value space, as the number of districts will grow with the size of the city, and therefore probably not solvable in real-time. We have therefore chosen to use an iterative approach, at the cost of not producing the optimum result. Possible areas where solutions for this type of layout optimization problem could be found are Pareto optimization algorithms [Knowles 2006, Deb et al. 2000] and Operations Research mathematics [Neumann and Morlock 1993, Mueller-Merbach 1992]. However, the question remains whether an “optimum result” in terms of district placement would improve the overall city layout, as our results already satisfy all the demands we had for our program.

# Chapter 8

# Conclusions

This chapter contains the conclusions that we made from the conducted work, sums up the contributions of new knowledge that the thesis makes, and discusses future research that can be done on the subject.

The problem stated in Chapter 3 has been solved:

1. As shown in Chapter 6, it is possible to procedurally generate the layout of a city in such a way that a non-expert in urban planning can generate believable city layouts without the use of external input data and using only a few, intuitive input parameters.
2. The principal mechanism for generating believable layouts lies in the integration of urban land use models from the field of urban geography into the generation process.
3. Layout generation can be achieved in near real time for cities of any (realistic) size.

## 8.1 Summary of Contributions

This thesis offers the following contributions:

1. *Analyzing existing work in procedural city generation and identifying areas that can be improved.* A study of the existing techniques for the procedural generation of urban structures led to the conclusion that, while many areas are far advanced, there is much room for improvement in the generation of realistic cities, especially

concerning the need for additional data in the form of e.g. floor plans, maps and population distribution and the required expertise of the users.

2. *Defining an approach to add believability to the procedural generation of cities.* This approach is anchored in scientific models of land use in cities, which describe the spatial relationships between different types of land use. The models are based on factors including the proximity between land use types and influences of the environment such as terrain and transportation veins.

3. *Developing an algorithm for the procedural generation of believable city layouts.* An algorithm which governs the distribution of different types of land use in a city was developed. It is based on the aforementioned land use models and was designed to maximise the generation speed.

4. *Implementation of a city layout generation tool.* A program was developed and integrated into a terrain framework software. It takes few, intuitive parameters as input and generates layouts of cities from different continents and with different historic backgrounds on a given terrain.

5. *Conducting interviews with experts to validate the used land use models and the implementation results.* Three researchers in the field of Urban Planning were consulted to guarantee valid results.

## 8.2 Future Research

The opportunities for future work in this area are diverse. They were discovered through limitations of the presented approach or they present interesting possibilities for further development.

1. *3D buildings.* The natural extension to this work is the generation of three dimensional buildings according to the specified land use types in a district. Care has to be taken to avoid uniform land use in a district, as the land use we assign to a district is the dominant land use in this area, but most districts are filled with a variety of building types, i.e. there will be some commerce in a residential district or some civic buildings in a mainly commercial area of the city core.

2. *Numerical solutions to optimize the district distribution.* We have chosen to avoid a numerical optimization algorithm to guarantee the fastest possible generation time for our city layouts. It would be very interesting to see how fast a numerical algorithm would handle this problem, which has a large value space over which

optimization has to occur. Possible areas where solutions can be found are Pareto optimization algorithms [Knowles 2006, Deb et al. 2000] and Operations Research mathematics [Neumann and Morlock 1993, Mueller-Merbach 1992].

3. *More varied models of land use.* We have implemented land use on only two continents, North America and Western Europe. Cities in other parts of the world, such as Asia or Africa might exhibit other structures that can be easily implemented using our algorithm by specifying their dominant land use types together with attraction and repulsion parameters. Additionally, there are other land use models for western cities that we did not model because they are less common or merely theoretical concepts of ideal cities, like the Garden City (Figure B.6). Those, too, could be modelled using our software.

4. *Mixed land use in a district.* The evaluation of our results led to comments of two urban design experts on how land use in a district is often not mono-functional, but a mix of different types of land use. This is often neglected by land use models, as exemplified by the fact that we did not come across a single model which proposes mixed use. It would increase the believability of generated cities if models could be implemented which describe the percentage of other land uses within a district, even if this district can be classified by its predominant land use type.

5. *Improved City Shape.* Our cities are mostly circular. This is a simplification and in reality many cities have diverse shapes. There are some suggestions that cities are shaped like fractals [Batty and Longley 1994]. This and other approaches can be included in the generation process to enhance the realism of the city outline. A simple way to make the edge of the city look more rugged is the use of Alpha Shapes [Akkiraju et al. 1995] to generate a city outline from the district centre points.

6. *Street networks.* Urban land use and other parameters of a city, such as the terrain, elevation or distance from the city centre, offer many clues about the type of street network pattern that will be encountered in a district. This information can be taken into account to fill districts with streets that exhibit organic, grid-like or other patterns.

# Appendix A

# Attraction and Repulsion Tables

## A.1 Attraction and Repulsion Tables

All the attraction value tables, described in Section 5.3.4

```
                case Districts.HeavyIndustry:
                   HeavyIndustry, 100);
                   Industry, 100);
                   HighclassResidential, -999
                   MiddleclassResidential, -50
                   WorkingclassResidential, 10
                   TransportationNodeGoods, 20
                   TransportationNodeHumans, -100
                   Commercial1, 0
                   Commercial2, 0
                   Market, 0
                   Church, 0
                   Civic, 0
                   Palace, 0
                   RoyalPalace, -200
                   CBDInnerCore, 0
                   CBDOuterCore, 0
                   CBDFrame, 0

               case Districts.Industry:
                   HeavyIndustry, 100
                   Industry, 100
                   HighclassResidential, -500
                   MiddleclassResidential, -50
                   WorkingclassResidential, 50
                   TransportationNodeGoods, 20
                   TransportationNodeHumans, -50
                   Commercial1, 0
                   Commercial2, 0
```

```
            Market, 0
            Church, 0
            Civic, 0
            Palace, -100
            RoyalPalace, -100
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.HighclassResidential:
            HeavyIndustry, -999
            Industry, -500
            HighclassResidential, 90
            MiddleclassResidential, 50
            WorkingclassResidential, -50
            TransportationNodeGoods, -100
            TransportationNodeHumans, -50
            Commercial1, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, 100
            RoyalPalace, 500
            Commercial2, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.MiddleclassResidential:
            HeavyIndustry, -100
            Industry, -80
            HighclassResidential, 100
            MiddleclassResidential, 90
            WorkingclassResidential, 50
            TransportationNodeGoods, -50
            TransportationNodeHumans, 10
            Commercial1, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, 0
            RoyalPalace, 0
            Commercial2, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.WorkingclassResidential:
            HeavyIndustry, 0
            Industry, 20
            HighclassResidential, 0
            MiddleclassResidential, 80
            WorkingclassResidential, 100
            TransportationNodeGoods, 10
            TransportationNodeHumans, 10
```

```
            Commercial1, 0
            Commercial2, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, 100
            RoyalPalace, -500
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.Commercial1:
            HeavyIndustry, -100
            Industry, -20
            HighclassResidential, 0
            MiddleclassResidential, 20
            WorkingclassResidential, 20
            TransportationNodeGoods, 0
            TransportationNodeHumans, 100
            Commercial1, 0
            Commercial2, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, 0
            RoyalPalace, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.Commercial2:
            HeavyIndustry, -20
            Industry, 0
            HighclassResidential, -50
            MiddleclassResidential, 0
            WorkingclassResidential, 20
            TransportationNodeGoods, 100
            TransportationNodeHumans, 0
            Commercial1, 0
            Commercial2, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, 0
            RoyalPalace, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.TransportationNodeGoods:
            HeavyIndustry, 100
            Industry, 100
            HighclassResidential, -100
            MiddleclassResidential, -80
            WorkingclassResidential, 0
```

```
            TransportationNodeGoods, −100
            TransportationNodeHumans, −50
            Commercial1, 0
            Commercial2, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, 0
            RoyalPalace, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.TransportationNodeHumans:
            HeavyIndustry, −50
            Industry, 0
            HighclassResidential, 0
            MiddleclassResidential, 80
            WorkingclassResidential, 100
            TransportationNodeGoods, −50
            TransportationNodeHumans, −100
            Commercial1, 0
            Commercial2, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, 0
            RoyalPalace, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.Church:
            HeavyIndustry, 0
            Industry, 0
            HighclassResidential, 0
            MiddleclassResidential, 0
            WorkingclassResidential, 0
            TransportationNodeGoods, 0
            TransportationNodeHumans, 0
            Commercial1, 0
            Commercial2, 0
            Market, 50
            Church, −200
            Civic, 0
            Palace, 0
            RoyalPalace, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.Market:
            HeavyIndustry, 0
            Industry, 0
            HighclassResidential, 0
```

```
            MiddleclassResidential, 0
            WorkingclassResidential, 0
            TransportationNodeGoods, 0
            TransportationNodeHumans, 0
            Commercial1, 0
            Commercial2, 0
            Market, -100
            Church, 50
            Civic, 0
            Palace, 0
            RoyalPalace, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.Palace:
            HeavyIndustry, 0
            Industry, 0
            HighclassResidential, 100
            MiddleclassResidential, 0
            WorkingclassResidential, 0
            TransportationNodeGoods, 0
            TransportationNodeHumans, 0
            Commercial1, 0
            Commercial2, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, -100
            RoyalPalace, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0

        case Districts.RoyalPalace:
            HeavyIndustry, -100
            Industry, -100
            HighclassResidential, 100
            MiddleclassResidential, 0
            WorkingclassResidential, -50
            TransportationNodeGoods, -100
            TransportationNodeHumans, -100
            Commercial1, 0
            Commercial2, 0
            Market, 0
            Church, 0
            Civic, 0
            Palace, -100
            RoyalPalace, 0
            CBDInnerCore, 0
            CBDOuterCore, 0
            CBDFrame, 0
```

Listing A.1: Attraction towards different land use types for all district types

```
            case Districts.HeavyIndustry:
                Dune, 100
                Ocean, -999
                Desert, -999
                Hill, -50
                Alpes, -999
                Dirt, 0
                Grass, 0
                HighMountain, -100
                MediumMountain, -70
                LowMountain, -50

            case Districts.Industry:
                Dune, 0
                Ocean, -999
                Desert, -999
                Hill, 0
                Alpes, -999
                Dirt, 0
                Grass, 0
                HighMountain, -100
                MediumMountain, -70
                LowMountain, -50

            case Districts.HighclassResidential:
                Dune, 20
                Ocean, -999
                Desert, -999
                Hill, 100
                Alpes, -999
                Dirt, -50
                Grass, 0
                HighMountain, -100
                MediumMountain, -70
                LowMountain, 10

            case Districts.MiddleclassResidential:
                Dune, 0
                Ocean, -999
                Desert, -999
                Hill, 20
                Alpes, -999
                Dirt, 0
                Grass, 0
                HighMountain, -100
                MediumMountain, -70
                LowMountain, -50

            case Districts.WorkingclassResidential:
                Dune, 0
                Ocean, -999
                Desert, -999
                Hill, 0
```

```
            Alpes, −999
            Dirt, 20
            Grass, 20
            HighMountain, −90
            MediumMountain, −70
            LowMountain, −60

        case Districts.TransportationNodeHumans:
            Dune, −50
            Ocean, −999
            Desert, −999
            Hill, −10
            Alpes, −999
            Dirt, 0
            Grass, 0
            HighMountain, −100
            MediumMountain, −100
            LowMountain, −100

        case Districts.TransportationNodeGoods:
            Dune, 20
            Ocean, −999
            Desert, −999
            Hill, −10
            Alpes, −999
            Dirt, 0
            Grass, 0
            HighMountain, −100
            MediumMountain, −100
            LowMountain, −100

        case Districts.Commercial1:
            Dune, −10
            Ocean, −999
            Desert, −999
            Hill, −20
            Alpes, −999
            Dirt, 0
            Grass, 50
            HighMountain, −100
            MediumMountain, −100
            LowMountain, −100

        case Districts.Commercial2:
            Dune, −10
            Ocean, −999
            Desert, −999
            Hill, −10
            Alpes, −999
            Dirt, 0
            Grass, 0
            HighMountain, −100
            MediumMountain, −100
            LowMountain, −100
```

```
            case Districts.Church:
                Dune, 0
                Ocean, -999
                Desert, -999
                Hill, 50
                Alpes, -999
                Dirt, 40
                Grass, 40
                MediumMountain, -100
                HighMountain, -100
                LowMountain, -100

            case Districts.Market:
                Dune, -50
                Ocean, -999
                Desert, -999
                Hill, -20
                Alpes, -999
                Dirt, 100
                Grass, 100
                HighMountain, -100
                MediumMountain, -100
                LowMountain, -100
```

Listing A.2: Attraction towards different terrain types for all district types.

```

             case Districts.HeavyIndustry:
                Core, -100
                Fringe, 20
                Suburb, 50
                Exurb, 100

            case Districts.Industry:
                Core, -100
                Fringe, 100
                Suburb, 50
                Exurb, 50

            case Districts.HighclassResidential:
                {
                    case Continent.WesternEurope:
                        Core, 00
                        Fringe, 100
                        Suburb, 0
                        Exurb, -100

                    case Continent.NorthAmerica:
                        Core, 0
                        Fringe, -100
                        Suburb, 00
                        Exurb, 100
                }
```

```
                case Districts.MiddleclassResidential:
                    Core, 50
                    Fringe, 100
                    Suburb, 100
                    Exurb, 100

                case Districts.WorkingclassResidential:
                    {
                        case Continent.WesternEurope:
                            Core, -100
                            Fringe, -100
                            Suburb, 0
                            Exurb, 100

                        case Continent.NorthAmerica:
                            Core, 100
                            Fringe, 0
                            Suburb, -100
                            Exurb, -200
                     }

                case Districts.TransportationNodeHumans:
                    Core, 0
                    Fringe, 100
                    Suburb, 100
                    Exurb, 100

                case Districts.TransportationNodeGoods:
                    Core, -100
                    Fringe, 50
                    Suburb, 100
                    Exurb, 0

                case Districts.Commercial1:
                    Core, 0
                    Fringe, 0
                    Suburb, 0
                    Exurb, 0

                case Districts.Commercial2:
                    Core, -100
                    Fringe, 50
                    Suburb, 100
                    Exurb, 100

                case Districts.Church:
                    Core, 100
                    Fringe, 50
                    Suburb, 0
                    Exurb, 0

                case Districts.Market:
                    Core, 100
                    Fringe, 50
                    Suburb, 0
```

```
            Exurb, 0

        default:
            Core, 0
            Fringe, 0
            Suburb, 0
            Exurb, 0
```

Listing A.3: Attraction towards different city areas for all district types.

```
        case Districts.HeavyIndustry:
            FarAway, −100
            InRiver, −999
            Riverbank, 100

        case Districts.Industry:
            FarAway, −100
            InRiver, −999
            Riverbank, 100

        case Districts.HighclassResidential:
            FarAway, 0
            InRiver, −999
            Riverbank, 50

        case Districts.MiddleclassResidential:
            FarAway, 0
            InRiver, −999
            Riverbank, 0

        case Districts.WorkingclassResidential:
            FarAway, 0
            InRiver, −999
            Riverbank, 0

        case Districts.TransportationNodeHumans:
            FarAway, 0
            InRiver, −999
            Riverbank, 0

        case Districts.TransportationNodeGoods:
            FarAway, 0
            InRiver, −999
            Riverbank, 0

        case Districts.Commercial1:
            FarAway, 0
            InRiver, −999
            Riverbank, 0

        case Districts.Commercial2:
            FarAway, 0
            InRiver, −999
```

```
                    Riverbank, 0

                case Districts.Church:
                    FarAway, 0
                    InRiver, -999
                    Riverbank, 0

                case Districts.Market:
                    FarAway, 0
                    InRiver, -999
                    Riverbank, 0
```

Listing A.4: Attraction towards different distances from a river for all district types.

```

                case Districts.HighclassResidential:
                    OnRoad, -100
                    Close, -100
                    FarAway, 100

                case Districts.MiddleclassResidential:
                    OnRoad, -80
                    Close, -80
                    FarAway, 80

                case Districts.WorkingclassResidential:
                    OnRoad, 0
                    Close, 0
                    FarAway, 50

                case Districts.Industry:
                    OnRoad, 100
                    Close, 100
                    FarAway, -100

                case Districts.HeavyIndustry:
                    OnRoad, 100
                    Close, 100
                    FarAway, -100

                case Districts.TransportationNodeHumans:
                    OnRoad, 100
                    Close, 100
                    FarAway, 0

                case Districts.TransportationNodeGoods:
                    OnRoad, 100
                    Close, 100
                    FarAway, 0

                case Districts.Commercial1:
                    OnRoad, 0
                    Close, 0
                    FarAway, 0
```

```
                case Districts.Commercial2:
                    OnRoad, 0
                    Close, 0
                    FarAway, 0
```

Listing A.5: Attraction towards different distances from a highway for all district types.

# Appendix B

# Land Use and City Shape

## B.1 Other spatial models of cities

### B.1.1 Descriptive Land Use Models

- Elkins: the German city (Figure B.1).
- Ashworth: the Dutch town (Figure B.1).
- Mann: the northern English city (Figure B.2).
- Robson: the British city (Figure B.2).
- Boustedt: the decentralized city, the city-region and the urban region (Figure B.3).
- Lichtenberger: the European city (Figure B.3).
- Neller: the urban agglomeration (Figure B.3).
- Buursink: the Dutch town (Figure B.4).

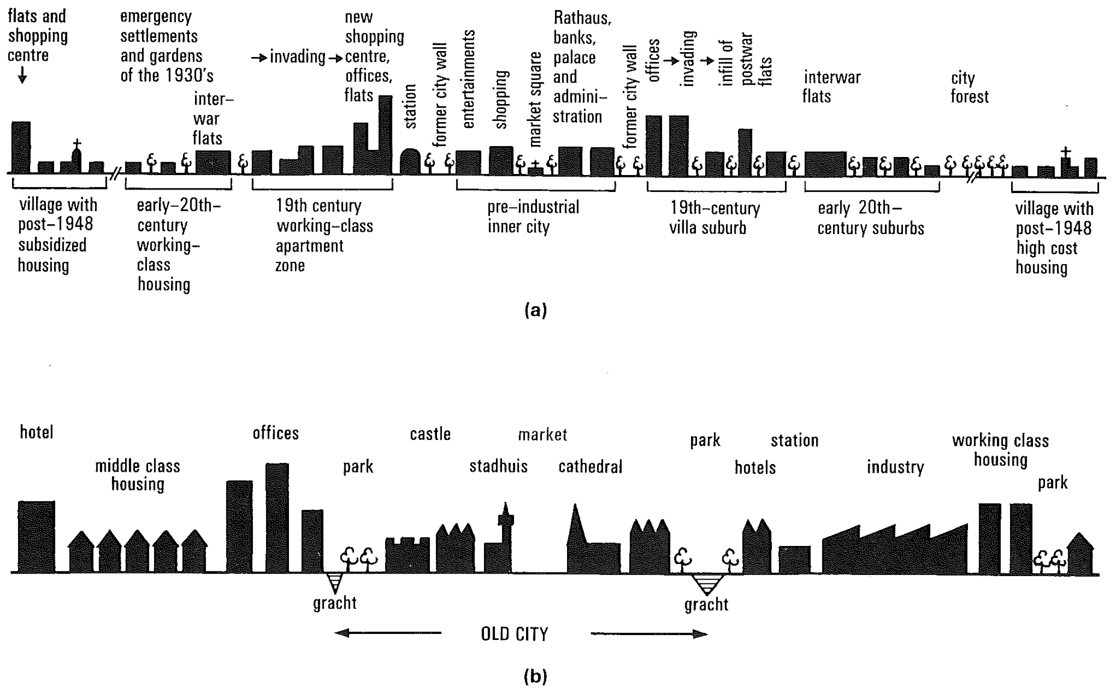


Figure 3.1 Descriptive models of West European cities: (a) Elkins' German city (reproduced from *The Geographical Magazine*, London); (b) Ashworth's Dutch town

FIGURE B.1: Elkins' German town and Ashworth's Dutch town. Source: [Burtenshaw et al. 1991].

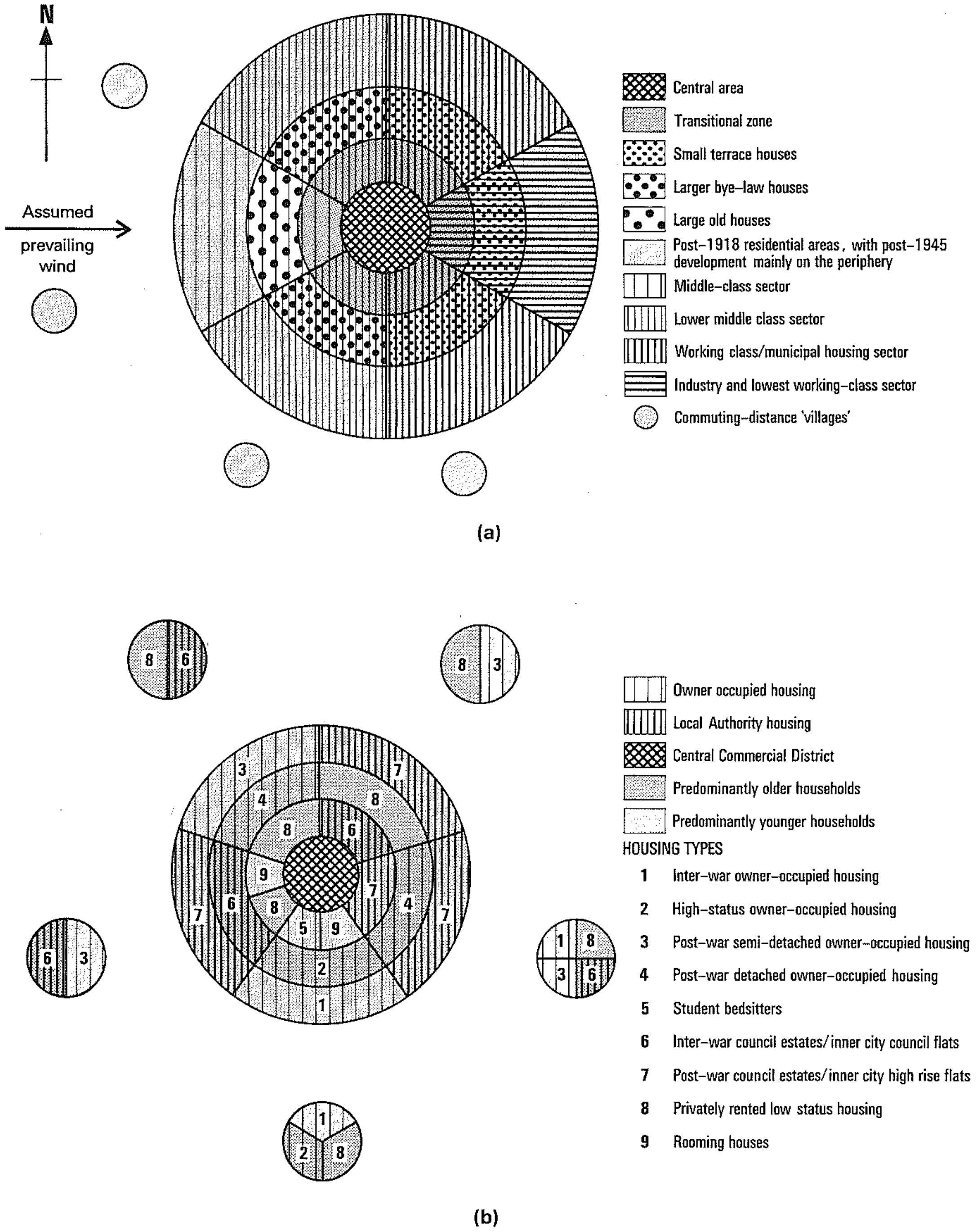


**Figure 3.2** Models of the English city: (a) Mann's northern English city; (b) Robson's British city (reproduced by permission of Oxford University Press)

FIGURE B.2: Land use in the English city by Mann and Robson. Source: [Burtenshaw et al. 1991].

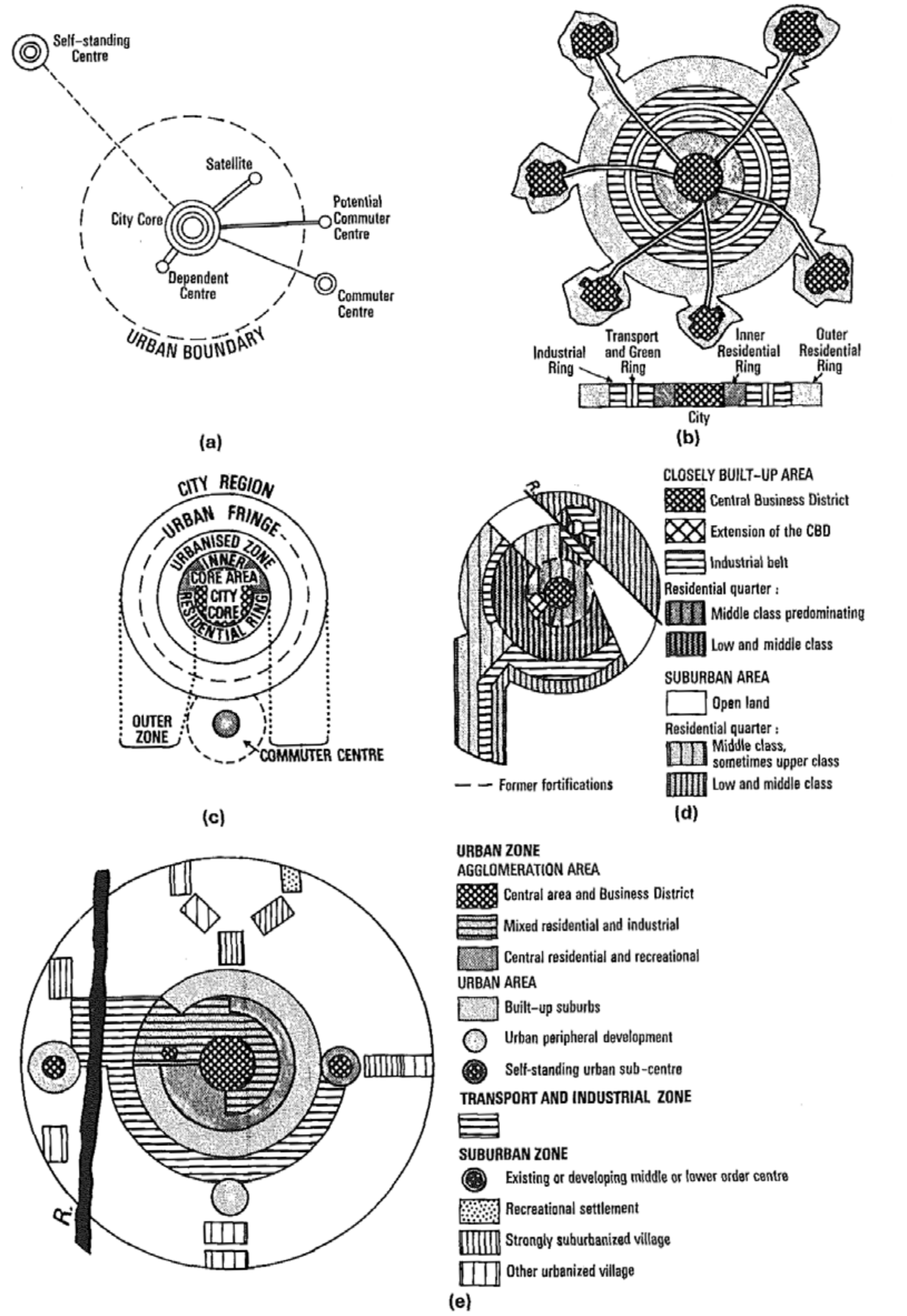


Figure 3.3 (a) Boustedt's decentralized city, 1967 (reproduced by permission of Akademie für Raunforschung und Landesplanung); (b) Boustedt's city-region, 1970; (c) Boustedt's urban region, 1970; (d) Lichtenberger's European city, 1970 (reproduced by permission of Pergamon Press); (e) Nellner's urban agglomeration, 1976

FIGURE B.3: Land use models by Boustedt, Lichtenberger and Neller. Source: [Burtenshaw et al. 1991].

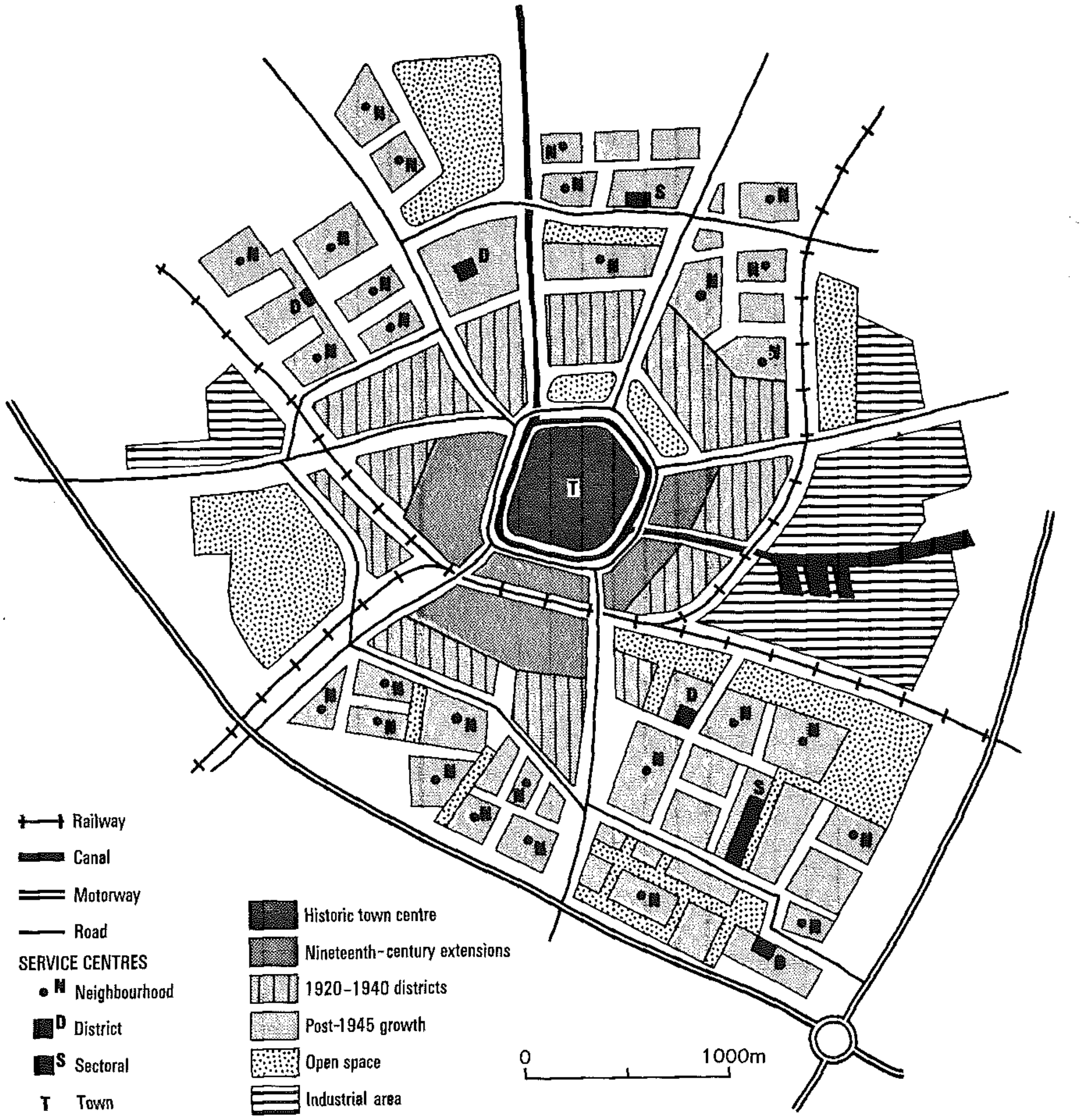


Figure 3.4 Buursink's Dutch town, 1977 (reproduced by permission of Rijksuniversiteit Groningen)

FIGURE B.4: Buursink's Dutch town. Source: [Burtenshaw et al. 1991].

### B.1.2 Utopian Land Use Models

- Figure B.5: LeCorbusier, “La Ville Radieuse”.
- Figure B.6: Ebenezer Howard, “The Garden City”.

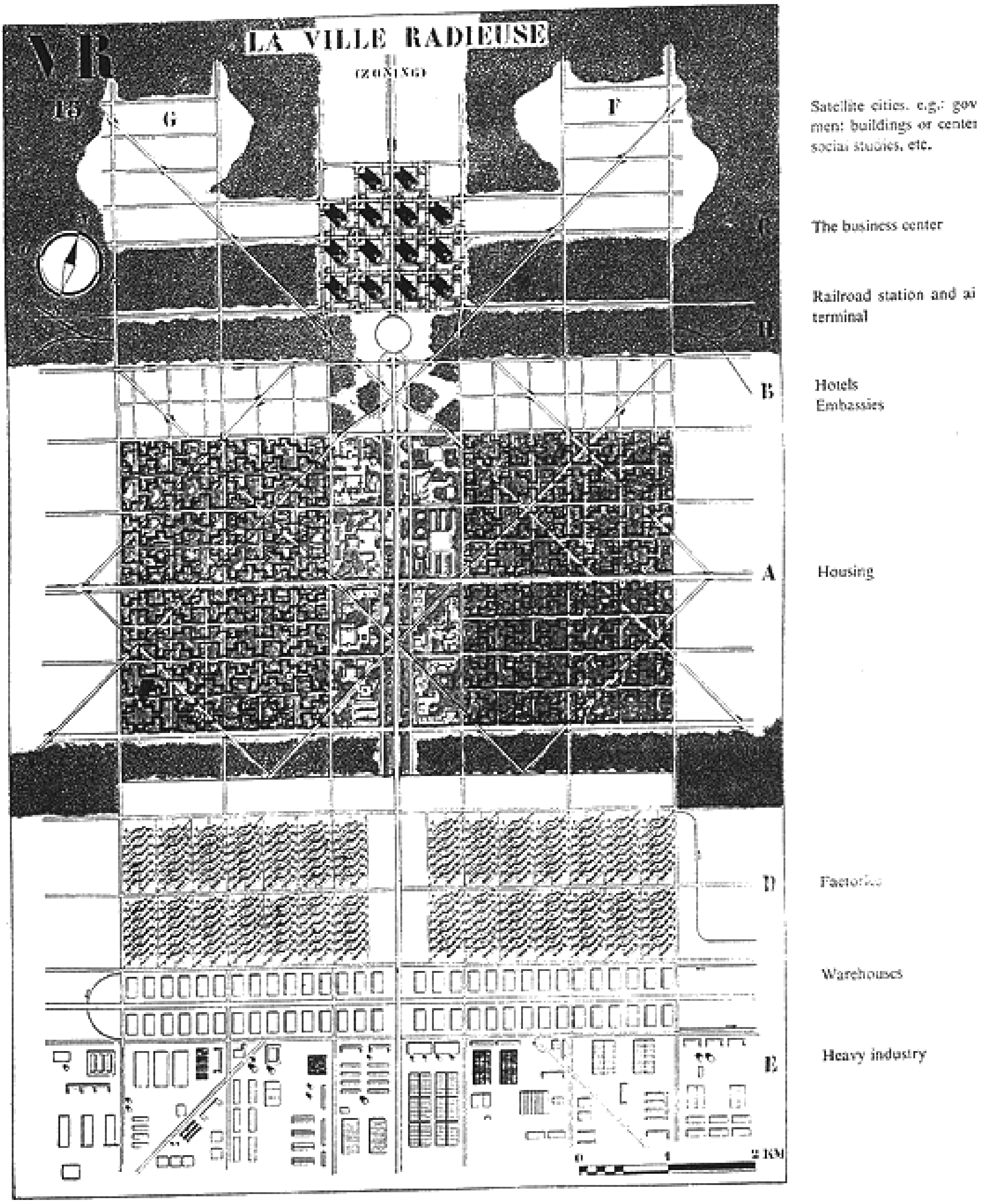


FIGURE B.5: LeCorbusier: La Ville Radieuse. Source: [Kennedy 1998]

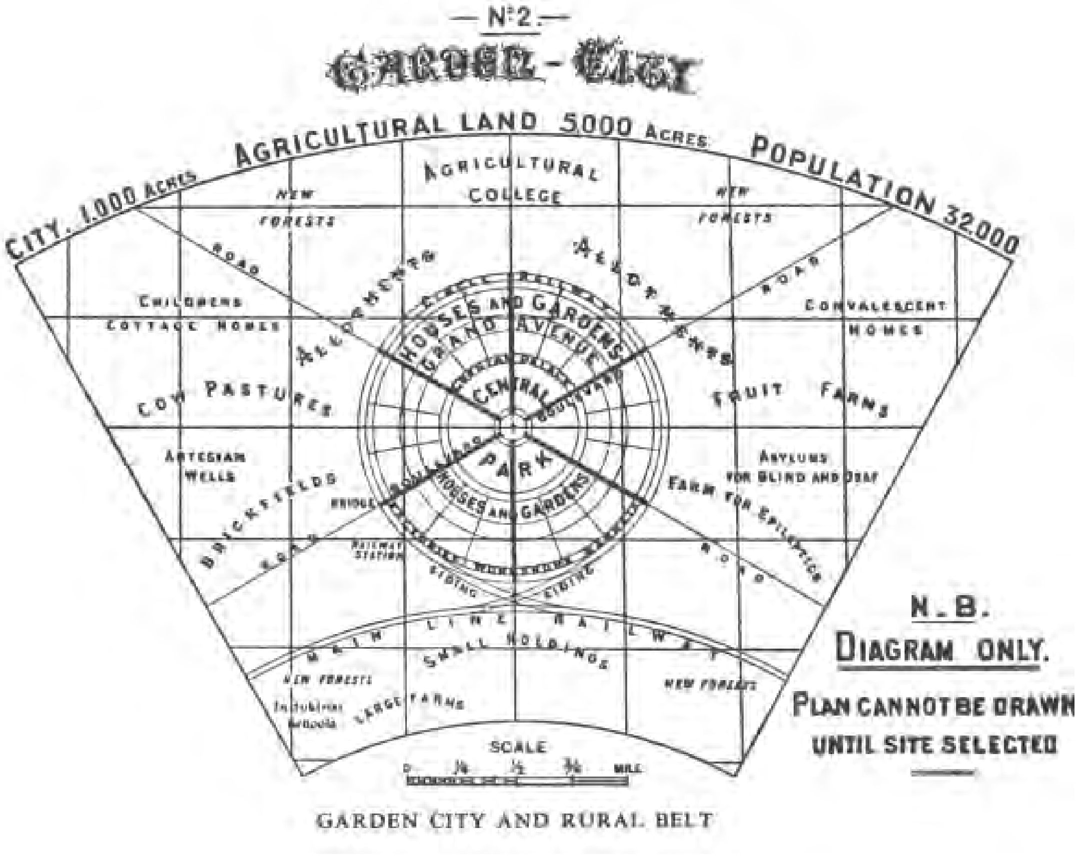


FIGURE B.6: Ebenezer Howard: The Garden City. Source: [Howard 1902]

## B.2 Examples for Historic City Cores

### B.2.1 Mercantile Historic Core

- Zwolle, The Netherlands (Figure B.7)
- Leuven, Belgium (Figure B.8)
- Neubrandenburg, Mecklenburg-Vorpommern, Germany (Figure B.9)

Figure B.7: Zwolle, The Netherlands. Mercantile historic core. Source: [Google 2009]

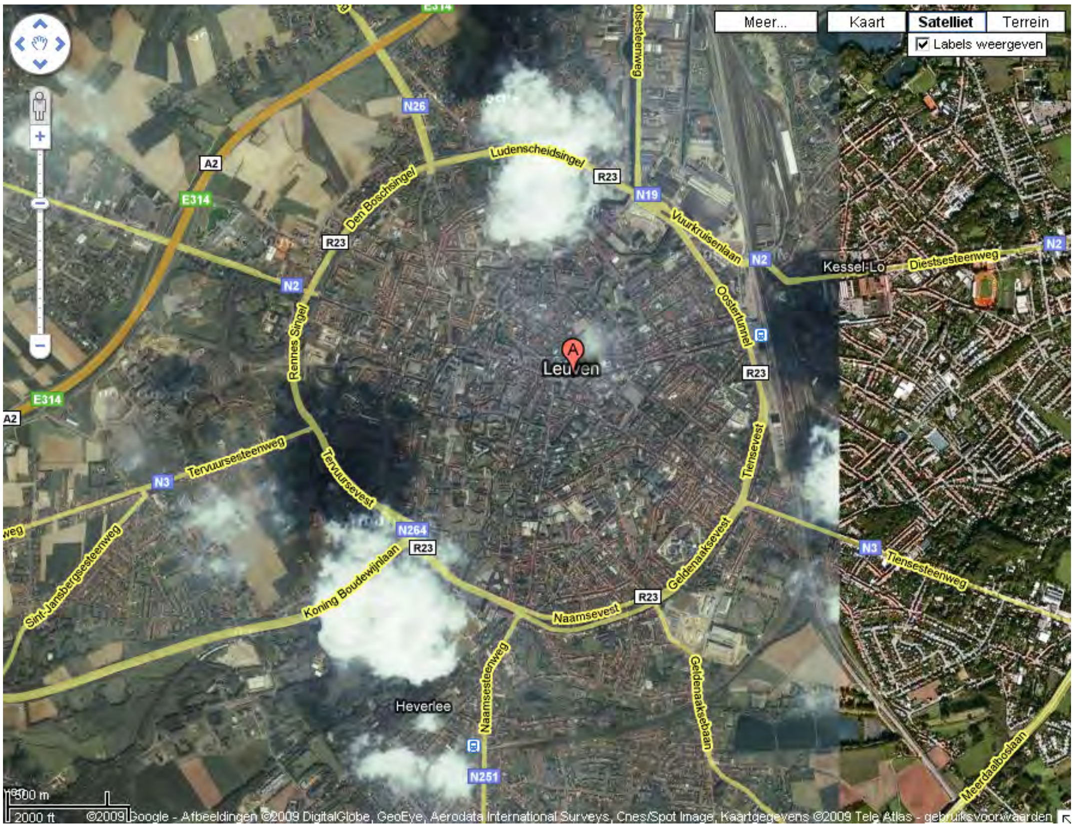


FIGURE B.8: Leuven, Belgium. Mercantile historic core. Source: [Google 2009]

FIGURE B.9: Neubrandenburg, Mecklenburg-Vorpommern, Germany. Mercantile historic core. Source: [Google 2009]

### B.2.2 Feudal Historic Core

- Madrid, Spain (Figure B.10)
- Toledo, Spain (Figure B.11)
- Siena, Italy (Figure B.12)
- San Gimignano, Italy

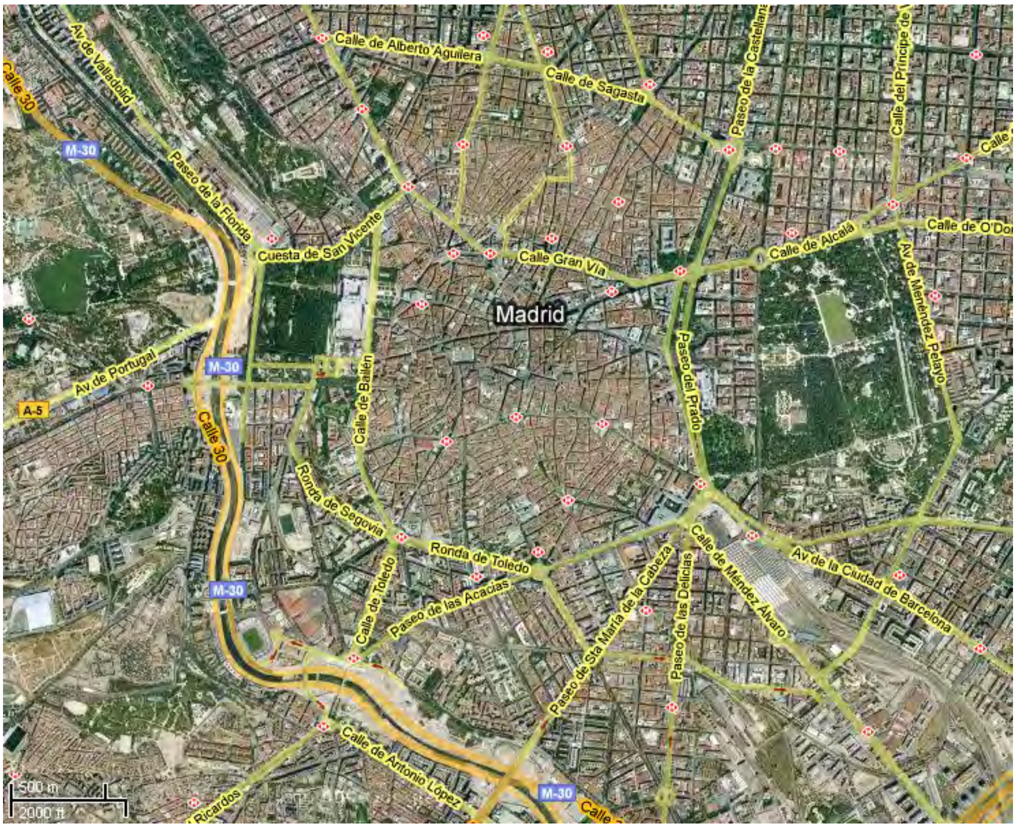


Figure B.10: Madrid, Spain. Feudal historic core. Source: [Google 2009].

FIGURE B.11: Toledo, Spain. Feudal historic core. Source: [Google 2009].

FIGURE B.12: Siena, Italy. Feudal historic core. Source: [Google 2009].

### B.2.3 Absolutistic Historic Core

- Stockholm, Sweden (Figure B.13)
- Versailles, France (Figure B.14)
- Karlsruhe, Germany

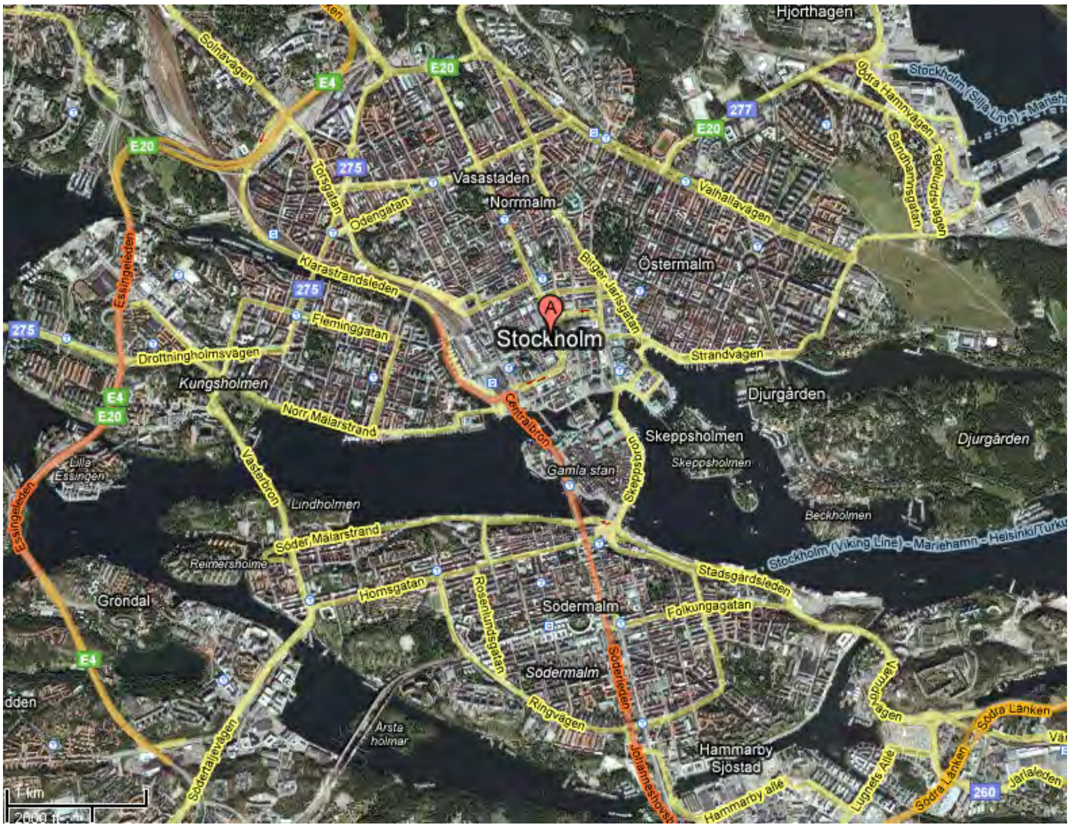


FIGURE B.13: Stockholm, Sweden. A city centered around the royal palace. Absolutistic historic core. Source: [Google 2009].

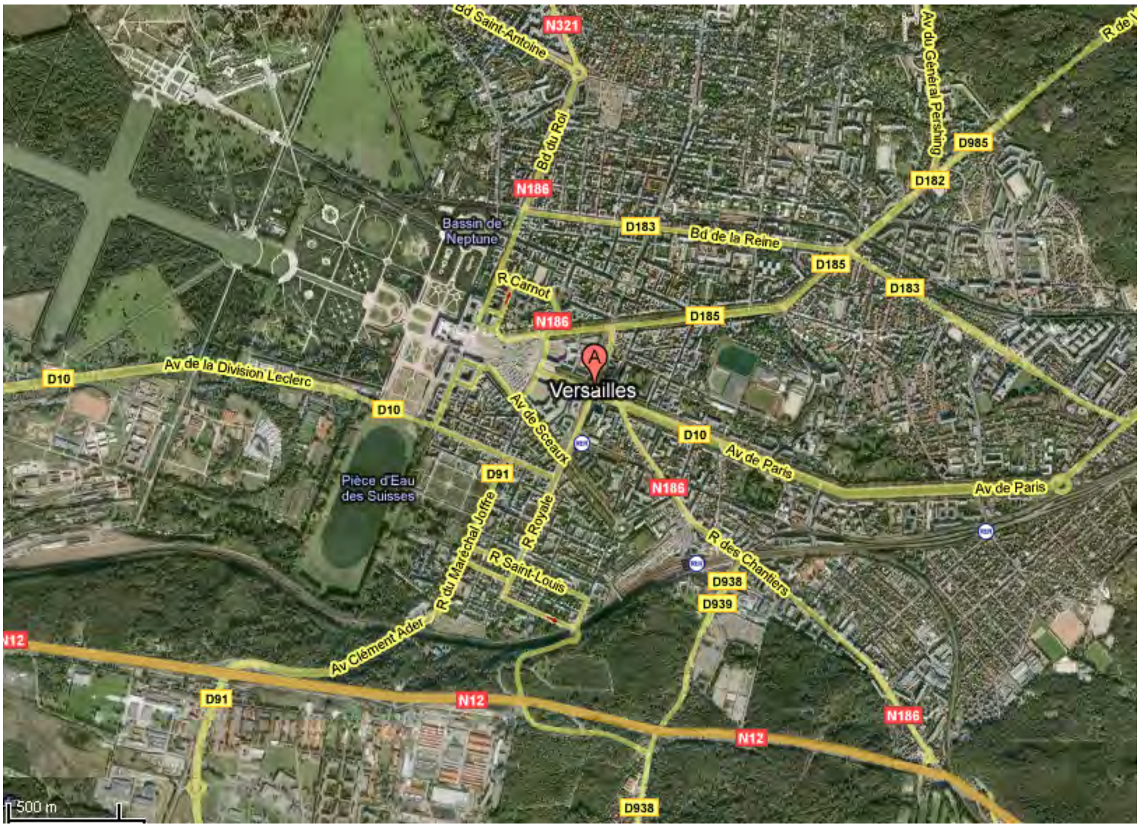


FIGURE B.14: Versailles, France. Symbol of absolute monarchy. Absolutistic historic core. Source: [Google 2009].

## B.3 Circular Cities

Figure B.15 to Figure B.20 show examples of circular cities from around the world.

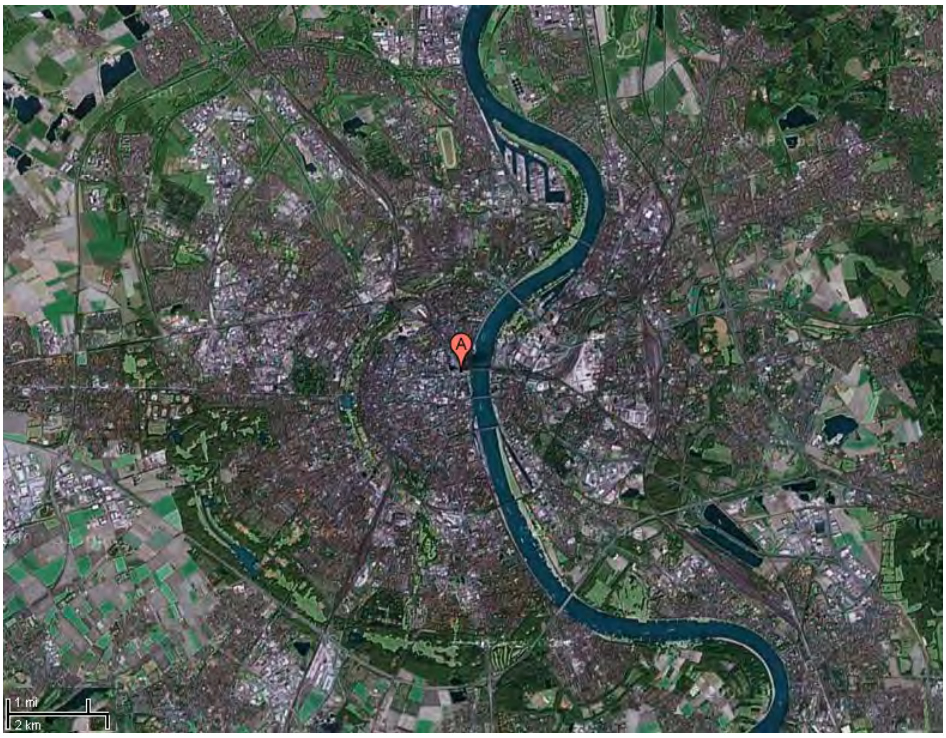


FIGURE B.15: Cologne, Germany. Source: [Google 2009].

FIGURE B.16: Amsterdam, The Netherlands. Source: [Google 2009].

Figure B.17: Palmanova, Spain. Source: [Google 2009].

Figure B.18: Nahalal, Israel. Source: [Google 2009].

Figure B.19: Hamadan, Iran. Source: [Google 2009].

Figure B.20: Firuzabad, Iran. Source: [Google 2009].

# Bibliography

Akkiraju, N., H., E., M., F., Fu, P., Mücke, E. P., and Varela, C. (1995). Alpha shapes: Definition and software. In *Proceedings of the 1st International Computational Geometry Software Workshop*, pages 63–66.

Alegre, F. and Dellaert, F. (2004). A probabilistic approach to the semantic interpretation of building facades. Technical report, Georgia Institute of Technology.

Alexander, C. (1999). The origins of pattern theory: The future of the theory, and the generation of a living world. *IEEE Softw.*, 16(5):71–82.

Alexander, C. (2003). New concepts in complexity theory arising from studies in the field of architecture. http://www.natureoforder.com/library/scientific-introduction.pdf.

Basilica dell'Osservanza (2009). La mappa dei palazzi di Siena. World Wide Web electronic publication. [Online; accessed 17-February-2009].

Batty, M. and Longley, P. (1994). *Fractal Cities: A Geometry of Form and Function.* Academic Press, San Diego, CA and London.

Benevolo, L. (1995). *The European City.* Blackwell.

Bethesda Softworks (2002). The Elder Scrolls III: Morrowind. computer game. http://www.elderscrolls.com/.

Bethesda Softworks (2006). The Elder Scrolls IV: Oblivion. computer game. http://www.elderscrolls.com/.

Birch, P. J., Browne, S. P., Jennings, V. J., Day, A. M., and Arnold, D. B. (2001). Rapid procedural-modelling of architectural structures. In *VAST '01: Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage*, pages 187–196, New York, NY, USA. ACM Press.

Blizzard Entertainment (1996). Diablo. computer game. http://www.blizzard.com/diablo/.

Brebion, F. (2009). Infinity. computer game (not yet released). http://www.infinity-universe.com.

Brenner, C. (2000). Towards fully automatic generation of city models.

Britton, I. (2001a). Glasgow Central Station. `http://www.freefoto.com`. [Online; accessed 05-March-2009].

Britton, I. (2001b). Holiday Inn Hotel, Southampton, Hampshire, UK. `http://www.freefoto.com`. [Online; accessed 05-March-2009].

Britton, I. (2001c). Manchester International Airport, the United Kingdom's third busiest airport. `http://www.freefoto.com`. [Online; accessed 05-March-2009].

Britton, I. (2001d). Meadowhall shopping centre, Sheffield, South Yorkshire, England. `http://www.freefoto.com`. [Online; accessed 05-March-2009].

Britton, I. (2001e). Picture of an industrial complex, Tesside, UK. `http://www.freefoto.com`. [Online; accessed 05-March-2009].

Britton, I. (2001f). Picture of industry by the River Tees near Middlesbrough, UK. `http://www.freefoto.com`. [Online; accessed 05-March-2009].

Britton, I. (2001g). Picture of the Port of Teesport on the River Tees near Middlesbrough. `http://www.freefoto.com`. [Online; accessed 05-March-2009].

Burgess, E. W. (1925). The growth of the city: an introduction to a research project. *The City*, pages 47–62.

Burtenshaw, D., Ashworth, G. J., and Bateman, M. (1991). *The City in West Europe*. Fulton, London.

Campbell, H. S. (1998). Residential Land Use. `http://www.geoearth.uncc.edu/faculty/hscampbe//landuse/b-models/D-res.html`. [Online; accessed 05-March-2009].

Capo, D. (2004). The fractal nature of the architectural orders. *Nexus Network Journal*, 6(1).

Carter, H. (1981). *The Study Of Urban Geography*. Edward Arnold; Co-published by Halsted Press, London; New York, 3rd edition.

Chen, G., Esch, G., Wonka, P., Müller, P., and Zhang, E. (2008). Interactive procedural street modeling. *ACM Trans. Graph.*, 27(3).

Cheng, J. and Masser, I. (2003). Urban growth pattern modeling: a case study of Wuhan city, PR China. *Landscape and Urban Planning*, pages 199–217.

Chomsky, N. (1956). Three models for the description of language. *Information Theory, IEEE Transactions on*, 2(3):113–124.

City of Manitowoc (1999). Existing Land Uses Profile. `http://www.manitowoc.org/compplan/ch3flanduse.pdf`. [Online; accessed 11-March-2009].

City of Newport News (2008). Existing Land Use. `http://www2.ci.newport-news.va.us/newport-news/plan/framework2008/section_d393749e304.html`. [Online; accessed 11-March-2009].

City of Sydney (2005). Land Use. `http://www.cityofsydney.nsw.gov.au/Environment/Land/CurrentStatus/LandUse.asp`. [Online; accessed 11-March-2009].

Clark, D. (1982). *Urban geography: an introductory guide*. Croom Helm, London; Canberra.

Coelho, A. F., de Sousa, A. A., and Ferreira, F. N. (2005). Modelling urban scenes for lbms. In *Web3D '05: Proceedings of the tenth international conference on 3D Web technology*, pages 37–46, New York, NY, USA. ACM Press.

Coombe, D. (1996). Induced traffic: what do transportation models tell us? *Transportation*, 23(1):83–101.

Cutler, B., Dorsey, J., Mcmillan, L., Mueller, M., and Jagnow, R. (2002). A procedural approach to authoring solid models. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 302–311, New York, NY, USA. ACM Press.

da Silveira, L. G. and Musse, S. R. (2006). Real-time generation of populated virtual cities. In *VRST '06: Proceedings of the ACM symposium on Virtual reality software and technology*, pages 155–164, New York, NY, USA. ACM.

de Berg, M., van Kreveld, M., Overmars, M., and Schwarzkopf, O. (2000). *Computational Geometry*. Springer. Voronoi Diagrams: The Post Office Problem. Chapter 7. pp. 147-163.

de Halleux, J. (2007). Quickgraph: A 100% c# graph library with graphviz support. `http://www.codeproject.com/KB/miscctrl/quickgraph.aspx`.

Deb, K., Agrawal, S., Pratab, A., and Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J. J., and Schwefel, H.-P., editors, *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pages 849–858, Paris, France. Springer. Lecture Notes in Computer Science No. 1917.

Doull, A. (2008). The death of the level designer: Procedural content generation in games. http://roguelikedeveloper.blogspot.com/2008/01/death-of-level-designer-procedural.html. [Online; accessed 17-February-2009].

Ebert, D. S., Musgrave, K. F., Peachey, D., Perlin, K., and Worley, S. (2002). *Texturing & Modeling: A Procedural Approach, Third Edition (The Morgan Kaufmann Series in Computer Graphics)*. Morgan Kaufmann.

Encyclopædia Britannica (2009). Industry — Eencyclopæia Britannica Online. `http://www.britannica.com/EBchecked/topic/287256/industry/`. [Online; accessed 05-March-2009].

Esch, G., Wonka, P., Müller, P., and Zhang, E. (2007). Interactive procedural street modeling. In *SIGGRAPH '07: ACM SIGGRAPH 2007 sketches*, New York, NY, USA. ACM.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining*, pages 226–231, Portland, OR.

Frischer, B. (2008). The Rome Reborn Project. How technology is helping us to study history. In *OpEd*. University of Virginia.

Fuller, A. (2007). Urban modeling in games. In *SIGGRAPH '07: ACM SIGGRAPH 2007 courses*, pages 148–166, New York, NY, USA. ACM.

geographyalltheway (2009). geographyalltheway.com - Online Geography Resources — GCSE / IGCSE Geography. `http://www.geographyalltheway.com/igcse_geography/population_settlement/settlement/urban_models.htm`. [Online; accessed 05-March-2009].

Glass, K. R., Morkel, C., and Bangay, S. D. (2006). Duplicating road patterns in south african informal settlements using procedural techniques. In *Afrigaph '06: Proceedings of the 4th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*, pages 161–169, New York, NY, USA. ACM Press.

Gonzles, R. R. and Somoza, J. (2004). Theories, models and urban realities. From New York to Kathmandu. *Dela*, 21:69–81.

Google (2009). Google maps. `http://maps.google.com/`. [Online; accessed 17-February-2009].

Greuter, S., Parker, J., Stewart, N., and Leach, G. (2003). Real-time procedural generation of 'pseudo infinite' cities. In *GRAPHITE '03: Proceedings of the 1st international* *conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 87–ff, New York, NY, USA. ACM Press.

Greuter, S., Stewart, N., and Leach, G. (2004). Beyond the horizon: Computer-generated, three-dimensional, infinite virtual worlds without repetition. In *Image Text and Sound Conference 2004*. RMIT University.

Groenewegen, S. A., Smelik, R. M., de Kraker, K. J., and Bidarra, R. (2009). Procedural city layout generation based on urban land use models. In *Eurographics 2009 Short Papers*. Eurographics Association.

Hahn, E., Bose, P., and Whitehead, A. (2006). Persistent realtime building interior generation. In *sandbox '06: Proceedings of the 2006 ACM SIGGRAPH symposium on Videogames*, pages 179–186, New York, NY, USA. ACM.

Harris, C. D. and Ullman, E. L. (1945). The nature of cities. *Annals of the American Academy of Political and Social Science*, pages 7–17.

Hart, J. C. (1994). On efficiently representing procedural geometry.

Howard, E. (1902). *Garden Cities of To-Morrow*. Faber and Faber.

Hoyt, H. (1939). *The Structure and Growth of Residential Neighborhoods in American Cities*. Federal Housing Administration, Washington, DC.

Iben, H. N. and O'Brien, J. F. (2006). Generating surface crack patterns. In *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 177–185, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Image Building Systems (2008). Commercial Centers. `http://www.imagebuildingsystems.com/commercial_centers.html`. [Online; accessed 05-March-2009].

Ingram, R. (2001). Building virtual worlds: A city planning perspective.

Introversion Software (2005). Darwinia. computer game. http://www.darwinia.co.uk/.

Jess (2006). Procedural house generation: A method for dynamically generating floor plans.

Jiang, Z. (2007). The road extension model in the land change modeler for ecological sustainability of IDRISI. In *GIS '07: Proceedings of the 15th annual ACM international symposium on Advances in geographic information systems*, pages 1–8, New York, NY, USA. ACM.

Kelly, G. and McCabe, H. (2006). A survey of procedural techniques for city generation. *ITB Journal*.

Kelly, G. and Mccabe, H. (2007). Interactive city generation methods. In *SIGGRAPH '07: ACM SIGGRAPH 2007 posters*, New York, NY, USA. ACM.

Kelly, T. (2008). Tuesday, October 14, 2008: phd. get set. go. `http://twak.blogspot.com/2008_10_01_archive.html`. [Online; accessed 11-March-2009].

Kennedy, R. (1998). LeCorbusier and the Radiant City contra true urbanity and the Earth. `http://www.uky.edu/Classes/PS/776/Projects/Lecorbusier/lecorbusier.html`.

Knights, S. (2009). Central Business District — Wikipedia, The Free Encyclopedia. World Wide Web electronic publication. [Online; accessed 17-February-2009].

Knowles, J. (2006). Parego: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *Evolutionary Computation, IEEE Transactions on*, 10(1):50–66.

Knox, P. L. (1994). *Urbanization: An Introduction to Urban Geography*. Prentice Hall.

König, R. and Bauriedel, C. (2004). Computer-generated city structures. In *Generative Art Conference*.

Kuehne, B. and Martz, P. (2007). *OpenSceneGraph Reference Manual v2.2*. Skew Matrix Software and Blue Newt Software.

Lammer, S., Gehlsen, B., and Helbing, D. (2006). Scaling laws in the spatial structure of urban road networks. *Physica A: Statistical Mechanics and its Applications*, 363(1):89–95.

Larive, M. and Gaildrat, V. (2006). Wall grammar for building generation. In *GRAPHITE '06: Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia*, pages 429–437, New York, NY, USA. ACM.

Laycock, R. G. and Day, A. M. (2003). Automatically generating large urban environments based on the footprint data of buildings. In *SM '03: Proceedings of the eighth ACM symposium on Solid modeling and applications*, pages 346–351, New York, NY, USA. ACM Press.

Lechner, T., Ren, P., Watson, B., Brozefski, C., and Wilenski, U. (2006). Procedural modeling of urban land use. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Research posters*, New York, NY, USA. ACM.

Lechner, T., Watson, B., Ren, P., Wilensky, U., Tisue, S., and Felsen, M. (2004). Procedural modeling of land use in cities. Technical report, Northwestern University.

Lechner, T., Watson, B., Wilensky, U., and Felsen, M. (2003). Procedural city modeling. *1st Midwestern Graphics Conference.*

Levinson, D. and Zhang, L. (2006). The evolution of transportation networks. In *NetSci 2006.*

Lufrano, F. (2006). Le 17 contrade del Palio. World Wide Web electronic publication. [Online; accessed 17-February-2009].

Mandelbrot, B. B. (1982). *The Fractal Geometry of Nature.* W. H. Freeman.

Martinet, A., Galin, E., Desbenoit, B., and Hakkouche, S. (2004). Procedural modeling of cracks and fractures. In *SMI '04: Proceedings of the Shape Modeling International 2004 (SMI'04)*, pages 346–349, Washington, DC, USA. IEEE Computer Society.

Masuo, K. (1999). A study of urban road network development from the viewpoint of development methods - a case study of Matsuyama City. *Memoirs of the Faculty of Engineering, Ehime University.*

Maxis (2008). Spore. computer game. http://www.spore.com/.

Mcbryde, M. J. (2005). Generation of office buildings in large scale virtual worlds. Master's thesis, Trinity University.

Mcmillen, D. P. and Smith, S. C. (2003). The number of subcenters in large urban areas. *Journal of Urban Economics*, 53(3):321–338.

Microsoft Corporation (2001). C # programming language. `http://msdn.microsoft.com/en-us/vcsharp/aa336809.aspx`.

Microsoft Corporation (2005). Visual Studio 2005. `http://msdn.microsoft.com/en-us/library/ms950416.aspx`.

Mortonson, C. and Cooper, A. (2007). Emergent geometry: procedural modeling through behavior. In *SIGGRAPH '07: ACM SIGGRAPH 2007 posters*, New York, NY, USA. ACM.

Moshirnia, A. V. (2006). The impact of procedural generation and modding on the participatory design of educational video games. In Uskov, V., editor, *Computers and Advanced Technology in Education.*

Mueller, P. (2007). Applied procedural modeling. In *SIGGRAPH '07: ACM SIGGRAPH 2007 courses*, pages 112–147, New York, NY, USA. ACM.

Mueller-Merbach, H. (1992). *Operations Research. Methoden und Modelle der Optimalplanung.* Vahlen.

Müller, P., Wonka, P., Haegler, S., Ulmer, A., and Van Gool, L. (2006). Procedural modeling of buildings. *ACM Trans. Graph.*, 25(3):614–623.

Müller, P., Zeng, G., Wonka, P., and Van Gool, L. (2007). Image-based procedural modeling of facades. *ACM Trans. Graph.*, 26(3).

Musgrave, K. F. (1999). Towards a synthetic universe. *IEEE Comput. Graph. Appl.*, 19(6):4–5.

Neumann, K. and Morlock, M. (1993). *Operations Research.* Hanser, Muenchen, Wien.

NVIDIA (2009). Nvidia compute unified device architecture (CUDA). `http://www.nvidia.com/cuda`.

Pacione, M. (2007). *Urban Geography: A Global Perspective.* Taylor & Francis, 2 edition.

Parish, Y. I. H. and Müller, P. (2001). Procedural modeling of cities. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 301–308, New York, NY, USA. ACM Press.

Pasian, A. (2009). Krauss-Maffei Wegmann: Twenty-First Century Simulation. `http://www.presagis.com/resources/customer_profiles/profiles/twenty_first_century_simulation/`. [Online; accessed 05-March-2009].

Peitgen, H.-O., Jürgens, H., and Saupe, D. (1993). *Chaos and Fractals: New Frontiers of Science.* Springer.

Procedural Inc. (2008a). City Engine. `http://www.procedural.com/cityengine/features.html`. [Online; accessed 17-February-2009].

Procedural Inc. (2008b). City Engine: Selected Pictures. `http://www.procedural.com/cityengine/pictures.html`. [Online; accessed 17-February-2009].

Procedural Inc. (2008c). City Engine Showcase: Procedural Pompeii. `http://www.procedural.com/cityengine/showcases/procedural-pompeii.html`. [Online; accessed 17-February-2009].

Procedural Inc. (2008d). City Engine Showcase: RomeReborn. `http://www.procedural.com/cityengine/showcases/rome-reborn.html`. [Online; accessed 17-February-2009].

Prusinkiewicz, P. and Lindenmayer, A. (1991). *The Algorithmic Beauty of Plants (The Virtual Laboratory).* Springer.

Ralston, B. A. and Barber, G. M. (1982). A theoretical model of road development dynamics. *Annals of the Association of American Geographers*, 72(2):201–210.

Rasmussen, T. F. (1966). The development of a planned plurinuclear city region: Greater Oslo. *Papers in Regional Science*, 16(1):105–116.

Richmond, P. and Romano, D. (2007). Automatic generation of residential areas using geo-demographics. In *3Dgeoinfo07: 2nd International Workshop on 3D Geo-Information: Requirements, Acquisition, Modelling, Analysis, Visualisation*, Delft, The Netherlands.

Roden, T. and Parberry, I. (2004). A structured methodology for procedural content creation. *Lecture Notes in Computer Science*, 3166:151–156.

Rodrigue, J.-P. (2006). *The Geography of Transport Systems.* Hofstra University, Department of Economics and Geography.

Rubenstein, J. M. (2004). *The Cultural Landscape: An Introduction to Human Geography.* Prentice Hall College Div.

Scaparra, M. and Church, R. (2005). A grasp and path relinking heuristic for rural road network development. *Journal of Heuristics*, 11(1):89–108.

Smelik, R. M., Tutenel, T., de Kraker, K. J., and Bidarra, R. (2008). A proposal for a procedural terrain modelling framework. In *Proceedings of the 14th Eurographics Symposium on Virtual Environments (EGVE).*

Snyder, J. M. and Kajiya, J. T. (1992). Generative modeling: a symbolic system for geometric modeling. In *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, volume 26, pages 369–378, New York, NY, USA. ACM Press.

Steinhaus, H. (1956). Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci*, 1:801–804.

Stiny, G. (1975). *Pictorial and Formal Aspects of Shape and Shape Grammars.* Birkhauser Verlag, Basel, Switzerland.

Sun, J., Yu, X., Baciu, G., and Green, M. (2002). Template-based generation of road networks for virtual city modeling. In *VRST '02: Proceedings of the ACM symposium on Virtual reality software and technology*, pages 33–40, New York, NY, USA. ACM Press.

Ulmer, A., Halatsch, J., Kunze, A., Müller, P., and Van Gool, L. (2007). Procedural design of urban open spaces. In *Proceedings of eCAADe 2007.*

Urban Properties (2001). Light Manufacturing or Retail. `http://www.urban-properties.net/properties/property_desc.aspx?id=15`. [Online; accessed 05-March-2009].

Wang, L., Hua, W., and Bao, H. (2007). Procedural modeling of residential zone subject to urban planning constraints. In *Lecture Notes in Computer Science*, pages 150–161. Springer Berlin / Heidelberg.

Wang, L., Hua, W., and Bao, H. (2008). Procedural modeling of urban zone by optimization. *Comput. Animat. Virtual Worlds*, 19(5):569–578.

Watson, B. (2007). Real and virtual urban design. In *SIGGRAPH '07: ACM SIGGRAPH 2007 courses*, pages 167–228, New York, NY, USA. ACM.

Waugh, D. (2005). *Geography: An Integrated Approach.* Nelson Thornes Limited, 3 edition.

Weber, B., Mueller, P., Wonka, P., and Gross, M. (2009). Interactive geometric simulation of 4d cities. *Computer Graphics Forum.*

Wheaton, W. C. (1974). A comparative static analysis of urban spatial structure. *Journal of Economic Theory*, 9(2):223–237.

Wheelan, G., Kelly, G., and Mccabe, H. (2008). Roll your own city. *DIMEA 2008.*

White, P. (1984). *The West European City : A Social Geography.* Longman, London; New York.

Whitted, T. and Kajiya, J. (2005). Fully procedural graphics. In *HWWS '05: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, pages 81–90, New York, NY, USA. ACM Press.

Wikipedia (2009a). Rostock — Wikipedia, The Free Encyclopedia. `http://en.wikipedia.org/w/index.php?title=Rostock`. [Online; accessed 17-February-2009].

Wikipedia (2009b). San Gimignano — Wikipedia, The Free Encyclopedia. `http://en.wikipedia.org/w/index.php?title=San_Gimignano`. [Online; accessed 17-February-2009].

Wikipedia (2009c). Stadtmauer (koeln) — Wikipedia, The Free Encyclopedia. `http://de.wikipedia.org/wiki/Stadtmauer_(Koeln)`. [Online; accessed 17-February-2009].

Wonka, P., Wimmer, M., Sillion, F., and Ribarsky, W. (2003). Instant architecture. *ACM Trans. Graph.*, 22(3):669–677.

Wyvill, B., van Overveld, K., and Carpendale, S. (2004). Rendering cracks in batik. In *NPAR '04: Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering*, pages 61–149, New York, NY, USA. ACM.

Xie, F. and Levinson, D. (2005). The topological evolution of surface transportation networks. Technical report, University of Minnesota: Nexus Research Group.

Yong, L., Congfu, X. U., Zhigeng, P., and Yunhe, P. (2004). Semantic modeling project: building vernacular house of southeast china. In *VRCAI '04: Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, pages 412–418, New York, NY, USA. ACM Press.

# Procedural City Layout Generation Based on Urban Land Use Models

S. A. Groenewegen[1,2,3], R. M. Smelik[1,2], K. J. de Kraker[1] and R. Bidarra[2]

[1]TNO Defence, Security and Safety, The Netherlands
[2]Delft University of Technology, The Netherlands
[3]Bauhaus-Universität Weimar, Germany

---

**Abstract**

*Training and simulation applications in virtual worlds require significant amounts of urban environments. Procedural generation is an efficient way to create such models. Existing approaches for procedural modelling of cities aim at facilitating the work of urban planners and artists, but either require expert knowledge or external input data to generate results that resemble real-life cities, or they have long computation times, and thus are unsuitable for non-experts such as training instructors. We propose a method that procedurally creates layouts of structurally plausible cities from high-level, intuitive user input such as city size, location and historic background. The resulting layouts consist of different kinds of city districts which are arranged using constraints derived from established models of urban land use. Our approach avoids the need for external expert engagement in the creation process, and allows for the generation of large city layouts in seconds, making it significantly faster than comparable agent-based software and thus supporting the needs of non-expert creators of virtual cities for many applications.*

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.5]: Computational Geometry and Object Modelling—, Simulation and Modelling [I.6.7]: Types of Simulation - Gaming—

---

## 1. Introduction and Related Work

Procedural modelling is especially suitable for urban virtual environments, which require a large amount of models and contain many repeating structures. Building those by hand, as is the traditional approach, is tedious work for humans, but fast and easy to generate with procedural methods. What is often lacking in procedurally generated cities is realistic structure, as creation focuses on building and street generation and rather than on believability. Realism must be added by the user and is not inherent in the generation algorithms, requiring expert knowledge or external data such as population density maps or existing city structures [KM06]. This is not suitable in situations where plausible structures are required, such as training and simulation, but where users lack knowledge in modelling believable city layouts.

Parish and Mueller's CityEngine [PM01] can create complex and detailed models of cities, but utilizes numerous input maps and statistical data from real world cities to achieve realism. The software is designed to facilitate the re-creation of existing cities from external data or the placement of buildings and roads by users with a city structure in mind, it does not generate structure by itself. Kelly and MCabe's software CityGen [KM07] generates road networks and fills them with three different types of buildings (down-town, suburban or industrial) which are distributed randomly. Statistical data from existing city patterns could be added to create more realistic cityscapes. In the agent-based approach CityBuilder [LRF*04,LRW*06], the generation of land usage for buildings is done via the interaction of a number of agents performing the role of urban developers. Three distinct types of land use are defined: residential, commercial and industrial. A small town layout of limited scale can be generated over a period of approximately 15 minutes not including the generation of building geometry or textures.

Our work aims at a fast and simple generation of believable city layouts without the need for external input data and with an intuitive user interface. Our goal is that users without prior knowledge in architecture or urban planning are able to generate realistic city layouts for use in their applications, e.g. virtual training environments, urban simulations or computer games.

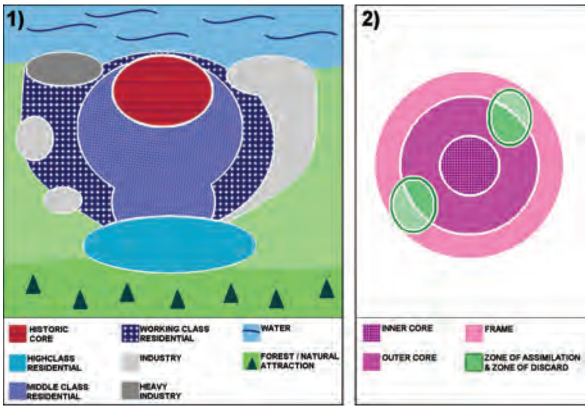


**Figure 1:** *1) A simple model of the Western European city. 2) Model of the Central Business District.*

## 2. Urban Land Use Models

The identification and explanation of patterns in the urban structure form a major topic in the field of urban geography. In a modern city, different districts are distinguishable by residential and social characteristics (i.e. industry, commerce, high-class and low-class residential areas, etc.). The distribution of those districts within a city is determined by a number of principles of urban land use. Simple generic models divide a city into rings or wedges, while more advanced models take into account different local and historic developments. As a basis for our algorithm, we have focused on land use models for two locations, Western Europe [Ben95, BAB91, Whi84] and North America [Kno94, Cla82, Car95].

**The Western European City.** Historically, the Western European city is characterized by landmarks such as castles, churches, monumental squares and other symbols of non-economic power. There exists a social and economic gradient from the center to the suburbs, with the wealthier population tending towards the city center. Figure 1.1 shows a general model of a West European city. The old city centers are sharply delineated from surrounding newer areas. Our model supports three different historic city cores:

1. Mercantile historic core: The mercantile city was prevalent in Northern Europe (e.g. the Hanseatic cities). Guilds grouped together all those involved in a particular trade, leading to districts dominated by different trades and little spatial segregation of different classes.
2. Feudal historic core: The feudal city, common in Southern Europe (e.g. Siena and San Gimignano), was a place where feudal families of different factions ruled a city. The (often warring) factions each built their own family palace, housing a noble family, which would dominate the district.
3. Absolutistic historic core: In subjugated cities (e.g. Karlsruhe or Stockholm), a single family or individual had the power to rebuild large sections of the city following detailed plans. This lead to large-scale geometric structures dominated by the royal palace.

High status districts are located in or close to the city center, especially in Southern Europe. Attractive surroundings can also lead to a high status district in the suburbs. The lower middle class can be found especially in suburban areas in post-war public housing. Some worker districts are in close proximity to the high status districts in the inner city. There is a general tendency for workers towards the periphery.

**The North American City.** In North America, city landmarks are most often skyscrapers and other symbols of economic power. There exist an anti-urban ethos which is manifested in a drift to the countryside of those who can afford the costs such as longer-distance commuting, leading to a social and economic gradient from the suburbs to the center. The layout of major roads has a big influence on the placement of districts, more so than in European cities. Characteristic for North American cities is the Central Business District (CBD) which makes up the city core. The CBD is the hub of economic, social and political life in the American city. The overall spatial structure of the CBD (Figure 1.2) is dominated by a high density core that contains the retail, office, entertainment and civic zones and a lower density frame with zones of warehousing, hotels, medical and education facilities and manufacturing.

## 3. Method

This section explains the steps involved in creating a city and the algorithm behind the automatic generation of the layout. Keep in mind that our approach does not intend to model the city growth process but aims at generating only the present state of a city. All parameters and their values are based on the urban land use models described in Section 2.

Generation is a three step process. First, the basic parameters of the city are set, then the districts of the city are generated and lastly their position within the city is determined. The location of the city is marked by designating an area on the terrain. Then the city is configured by setting several parameters. We chose the parameters so that users can easily convey which type of city to generate, as for example:

- city size (as diameter)
- continent on which the city is located (determining the way districts are distributed)
- historical background of the city, usually connected to its location (determining the districts in the core)
- number of highways passing through (influencing the district placement especially in North America)

The appropriate mix of district types in the city is determined based on the parameters. Our model currently includes 18 different district types: 3 residential districts (high class, middle class and working class - drawn in *light to dark blue*

in the output layouts), 2 industry districts (light and heavy industry - *grey*), 2 commercial districts (*green*), transportation nodes for humans and for goods (*yellow*), green spaces and 8 special core districts, e.g. with cathedrals or town squares (*other colours*).
Once all districts of a city are specified, their location within the city has to be determined. The district placement is influenced by five different factors: type of neighbouring districts (e.g. the high-end residences are unlikely to be situated near an industrial area), terrain type (e.g. industry districts preferring to be near water and highclass residential districts preferring hilly terrain), area of the city (e.g. to represent the social and economic gradient from the center to the suburbs), location of rivers and location of highways. Following the urban land use models, each district type is assigned different values for each of those factors to signify the degree of attraction or repulsion. For an example, the heavy industry districts have a high attraction towards waterside terrain, a high repulsion regarding high-class residential districts but a moderate attraction towards other industry.

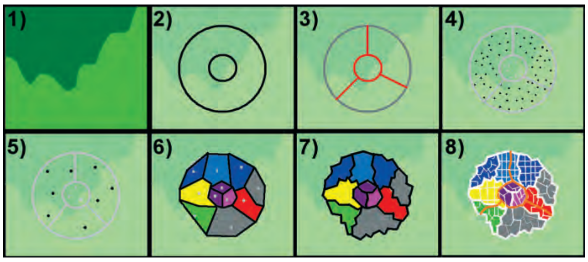


**Figure 2:** *Creation of a City. 1) Terrain 2) City Limits 3) Preliminary Highways 4) Candidate Locations 5) District Locations 6) Voronoi Graph 7) Noise 8) Streets*

The district placement algorithm (see below) is based on these attraction and repulsion coefficients. Figure 2 shows the progressive placement of districts in the city. First the city size and location on the terrain are defined by the user (2). The (preliminary) highways are drawn automatically around the core and leading out of the city, possibly towards other cities (3). For the placement of the districts, a number of candidate locations are generated based on a random distribution (4) and the best locations chosen for placement of the districts (5). After the locations of the districts have been determined, a Voronoi diagram is generated from the location points, leading to a polygon shape around each location which describes the district limits in such a way as to equally divide the terrain among the districts (6). Some noise is then added to the diagram to achieve a more realistic look (7) and a street network is generated according to known methods [PM01] (8). We have observed that the optimum number of candidate locations equals approximately twice the number of districts. A higher number of candidate locations does not yield better results.

**District placement algorithm**. The suitability $S$ of a location $l$ to accommodate a given district $d$ indicates their mutual fitness degree, taking into account the city parameters chosen for that urban land use model. For each district $d$ that has to be allocated, we compute the suitability $S$ of each available candidate location $l$, and assign to $d$ the location with the highest $S$. $S$ is a function of five parameters:

- $S_d$: placement of the district relative to the $n$ other already placed districts
- $S_t$: terrain type
- $S_a$: area within the city
- $S_r$: distance from rivers
- $S_h$: distance from highways

To give an example, equation 1 shows the calculation of $S_d$ - the suitability of a location regarding the $n$ surrounding already placed districts - , with $A_{d_i}$ being the attraction value towards a type of district $d_i$, $\Delta_{d_i}$ being the distance to $d_i$ and $\Delta_{min}$ the minimum possible distance between district centres. Attraction coefficients are assigned integer values between -100 (high repulsion) and 100 (high attraction), which depend on the chosen city type (e.g. European with mercantile core), while distances are measured in pixels.

$$S_d = \sum_{i=1}^{n} S_{d_i} = \sum_{i=1}^{n} A_{d_i} * 1/(\Delta_{d_i} - \Delta_{min}) \tag{1}$$

The values for each parameter are then weighted ($w$) according to their importance for the city type (e.g. distance from highways plays a greater role in American cities). The final value for $S$ at a candidate location $l$ is then defined as sum of all five values:

$$S = w_d * S_d + w_t * S_t + w_a * S_a + w_r * S_r + w_h * S_h \tag{2}$$

The location $l$ with the highest suitability $S$ for a district $d$ is thus chosen and the district is placed there. This is repeated until all districts have been placed. The procedure is first performed for all the districts of the city core and then for those in the rest of the city. Within both sets, the placement order of the districts is randomized to generate varying output layouts.

## 4. Results

Figure 3.1 shows a result for a European city with historic core, and Figure 3.2 shows a resulting city layout for a North American city with a CBD. They have visible similarities to the exemplary models (Fig. 1). The European city is a good example for the influence of the terrain: the highclass residential districts (*light blue*) are found on hilly terrain while the industry (*grey*) is at the seaside and the city perimeter is shaped by the presence of nearby mountains. The American city shows a discernible influence of the highways, with the industrial (*grey*) and commercial (*green*) districts as well as the transportation nodes (*yellow*) settling close to them, and the residential areas (*blue*) further away from them.

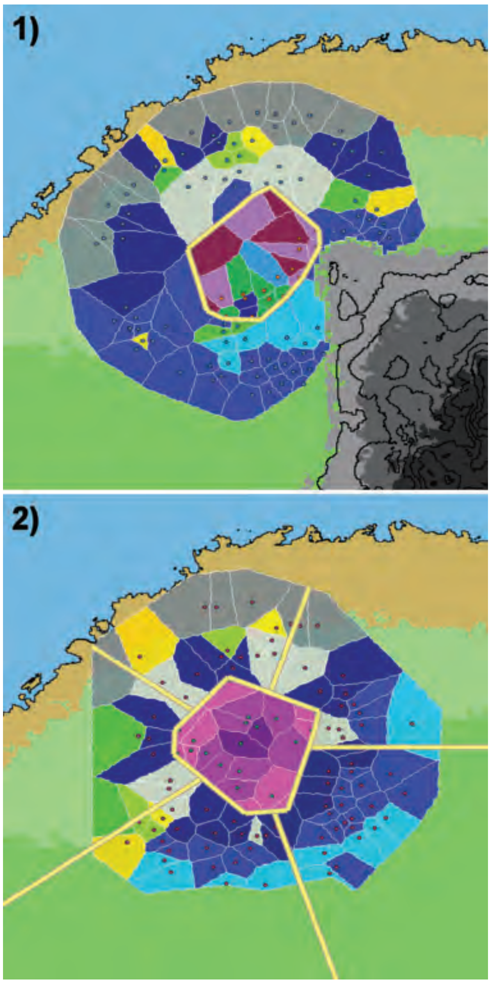


**Figure 3:** *1) A resulting layout for a Western European city, 2) A resulting layout for a North American city, shown with preliminary highways*

To validate our modelling approach, we have consulted experts on urban design and planning. They approved the city models and parameters we use in our algorithm and confirmed the plausibility of the output layouts.

Our algorithm efficiently avoids lengthy recursive iterations, yielding plausible output models conforming to scientific models of urban land use. Our system can generate a large city in a few seconds on a modern PC.

## 5. Conclusion

Our approach generates believable layouts of cities consisting of many different kinds of districts. The generation method is based on models of urban land use, thus requiring neither user knowledge on urban design and planning, nor external input data in order to achieve plausible city layouts. Several intuitive parameters allow choosing among several different types of cities based for example on their historic background, location, size and shape, covering therefore a large percentage of typical cities in the respective continental settings. Our efficient approach generates layouts in a few seconds, a clear performance distinction from much slower agent-based approaches.

In the future, we would like to extend our algorithm to include more locations (such as cities in Asia and Africa), more lifelike road networks among cities, and a broader choice of districts. Additionally, it would be very interesting to apply our method to generate less common or conventional city designs using different land use models, such as the linear city, garden city or the modernist city. Based on our layouts, street networks and 3D building models can be created using existing techniques for procedural generation.

This city generation algorithm is integrated within a prototype modeling framework aimed at procedurally generating virtual worlds, including terrain, vegetation and urban environments [STdKB08]. We believe that such an integrated framework will significantly improve the utility and deployment of training and simulation applications.

This research has been supported by the GATE project, funded by the Netherlands Organization for Scientific Research (NWO) and the Netherlands ICT Research and Innovation Authority (ICT Regie).

## References

[BAB91] Burtenshaw D., Ashworth G. J., Bateman M.: *The City in West Europe*. Fulton, London, 1991.

[Ben95] Benevolo L.: *The European City*. Blackwell, 1995.

[Car95] Carter H.: *The Study Of Urban Geography*, 4th ed. Edward Arnold, London, 1995.

[Cla82] Clark D.: *Urban geography: an introductory guide*. Croom Helm, London; Canberra, 1982.

[KM06] Kelly G., McCabe H.: A survey of procedural techniques for city generation. *ITB Journal* (2006).

[KM07] Kelly G., Mccabe H.: Interactive city generation methods. In *ACM SIGGRAPH 2007 posters* (2007), ACM.

[Kno94] Knox P. L.: *Urbanization: An Introduction to Urban Geography*. Prentice Hall, 1994.

[LRF*04] Lechner T., Ren P., Felsen M., Wilensky U., Tisue S., Watson B.: *Procedural modeling of land use in cities*. Tech. Rep. NWU-CS-04-38, Dept. of Computer Science, Northwestern University, 2004.

[LRW*06] Lechner T., Ren P., Watson B., Brozefski C., Wilenski U.: Procedural modeling of urban land use. In *ACM SIGGRAPH 2006 Research posters* (2006), ACM, p. 135.

[PM01] Parish Y. I. H., Müller P.: Procedural modeling of cities. In *: Proceedings of ACM SIGGRAPH 2001* (2001), ACM Press, pp. 301–308.

[STdKB08] Smelik R. M., Tutenel T., de Kraker K. J., Bidarra R.: A Proposal for a Procedural Terrain Modelling Framework. In *Poster Proceedings of the 14th Eurographics Symposium on Virtual Environments EGVE08* (2008).

[Whi84] White P.: *The West European City : A Social Geography*. Longman, London; New York, 1984.