A Multi-Scale Flexible Framework for Urban Modelling

James Virgo

Doctor of Philosophy

School of Engineering

July 2020

Abstract

The configuration of urban areas, and of infrastructures which serve them is central to managing the urbanisation process. Integrated assessment frameworks aim to inform decisions regarding planning, policy, and design to coordinate projects across sectors. Development of such models poses a number of challenges; (i) scenario generation, (ii) intelligibility to stakeholders, (iii) validity, (iv) control and feedback, (v) execution time, (vi) data requirements, (vii) uncertainties and, (viii) flexibility/reusability.

This research has developed a multi-scale flexible framework which disaggregates projected regional employment to ward-level population, and further to rasterised development. This comprises; (i) transport network generalised cost, (ii) cost composition, (iii) spatial interaction incorporating transport accessibility, (iv) development zoning, (v) multi-criteria evaluation of development suitability, and (vi) cellular development. The framework is generically implemented, each model being specified in terms of inputs, outputs, and parameters. Model-linkage is via input/output chaining, providing the opportunity to experiment with alternative solutions. Execution is flexible/configurable to perform multiple model runs whilst varying parameters and propagating metadata through stages. Python controls execution flow, C++ provides performance, PostgreSQL manages data, and QGIS assists input/output.

The framework is deployed in baseline scenarios for London and Innsbruck, and in more detailed scenario/uncertainty exploration for London. The framework's utility is judged by criteria corresponding to the above challenges and is found to be favourable, with performance, flexibility and uncertainty support as key attributes. The framework executes models for London in ~52 seconds on modest hardware (1.6GHz, 8GB). This involves cost-weighted Dijkstra - 4 transport networks (~42s), cost composition and accessibility conversion (~4s), spatial interaction - 633 wards (~2s), rasterised 4-hectare development zones (~1s), 7 criteria development suitability evaluation (~1s), and cellular development - 100m scale (~2s). Combinatorial uncertainties are accommodated by a flexible, modular structure which promotes reuse, and records run configuration as well as model parameters in chained metadata.

Acknowledgements

I would like to thank my supervisors Richard Dawson, Stuart Barr and Alistair Ford for their insight, encouragement and understanding during my PhD. I would also like to thank my family and friends for their support and patience.

The land use and transport models presented in this research draw upon and further develop techniques described in the Urban Integrated Assessment Framework (UIAF) for which I acknowledge the work of my supervisors Richard Dawson, Stuart Barr and Alistair Ford. The work of Richard Dawson, Stuart Barr, Alistair Ford and Tom Holderness on linking Python models with datasets stored in Postgres tables for pre-processing input transport networks is partly retained, adapted and enhanced in this thesis to build and store transport networks.

Data for the case study presented in Chapter 5 was provided by Christian Mikovits.

Table of Contents

List of Figures

List of Tables

List of Acronyms

AAF: Accessibility Assessment Framework

ABM: Agent-based Model

API: Application Programming Interface

BRT: Bus Rapid Transit

CA: Cellular Automata

CAS: Census Area Statistics

CSV: Comma-separated Values

DEM: Digital Elevation Model

DLL: Dynamic-link Library

E3MG: Global Energy-Environment-Economy Model

GHG: Greenhouse Gas

GIS: Geographic Information System

GLA: Greater London Authority

GLUD: Generalised Land Use Database

GPU: Graphics Processing Unit

GUI: Graphical User Interface

HTC: High Throughput Computing

IAM: Integrated Assessment and Modelling

LUTM: Land Use Transport Model

MC: Monte Carlo

MCE: Multi-criteria Evaluation

MDM: Multi-sectoral Dynamic Model of the UK

NOMIS: National Online Manpower Information System

ONS: Office for National Statistics

OS: Ordnance Survey

OSM: OpenStreetMap

PDF: Probability Density Function

PMI: Python Model Interface

QGIS: Quantum Geographic Information System

RUT: Random Utility Theory

SA: Sensitivity Analysis

SIC: Standard Industrial Classification

SIM: Spatial Interaction Model

SQL: Structured Query Language

SRS: Spatial Reference System

SSSI: Sites of Specific Scientific Interest

SWIG: Simplified Wrapper and Interface Generator

TAM: Transport Accessibility Model

TOD: Transit Oriented Development

UA: Uncertainty Analysis

UDM: Urban Development Model

UHI: Urban Heat Island

UIAF: Urban Integrated Assessment Framework

UIMF: Urban Integrated Modelling Framework

VBA: Visual Basic for Applications

ZM: Zonal Model

Chapter 1. Introduction

1.1 Urbanisation and Climate Change

The majority of the world's population now live in urban areas and the trend towards urbanisation is set to continue; global population growth projections indicate that between 2000 and 2050 the capacity of urban areas will need to be doubled in the case of developed countries and increased by over 300% in developing nations (UN, 2012).

This global urbanisation drives the expansion of existing urban areas, increasing the demands on land use, resources and infrastructure. Cities account for around 80% of global economic function and roughly 75% of global resource consumption, comprising energy and material flows. The current reliance on mostly finite resources to fuel economic growth places the process of urbanisation at the forefront of sustainability research. This focusses on decoupling economic growth from the increased use of resources, and reconfiguring urban infrastructure to provide more efficient intra-urban resource flows (Hodson *et al*., 2012).

Critical infrastructures underpin the economic and social function of modern society and rapid urbanisation in developing countries requires the construction of infrastructure at an unprecedented rate to meet the basic living demands of the increasing urban population (McKinsey, 2011). Inadequate infrastructure provision in these scenarios leads to the degradation of living and sanitation conditions, and the impedance of economic and social function (McKinsey, 2010).

Most of the world's urban areas with population greater than 5M inhabitants are situated within 100 km of the coast (Nicholls *et al*., 2007); this near-coastal region is estimated to house around a quarter of the world's population at a density nearly three times the global average (Small and Nicholls, 2003). The spatial expansion and densification of these areas increases exposure to the impacts of climate change; projected rises in sea level will lead to increased flooding and submergence in coastal zones. Poverty increases vulnerability to climate-related hazards which exacerbates other problems resulting from uneven development processes. By limiting the magnitude of climate change the overall risks of severe and irreversible impacts can be reduced (IPCC, 2014).

The Paris Agreement (UNFCC, 2015), commits signatories to limit anthropogenic climate change requiring a rapid reduction in greenhouse gas (GHG) emissions to slow the global rate of temperature increase. This mitigation effort must be coordinated with adaptation challenges posed by increasingly frequent and severe weather events accompanying existing climate change. Further to the Paris Agreement signed by national governments, city-level measures to reduce GHG emissions and adapt to climate change have been agreed by many of the world's city leaders (UCLG, 2015).

Cities consume resources and generate greenhouse gas emissions disproportionately to their spatial extent, they are also especially vulnerable to disrupted resource supplies and the effects of climate change (McEvoy *et al*., 2012).

The most likely critical effects of climate change on cities include:

- Impacts on energy consumption for heating and cooling.
- Impacts on water availability.
- Impacts of rising sea levels and storm surges on coastal areas.
- Impacts on built infrastructure of extreme events such as storms, flooding, heat extremes and droughts.
- Impacts on health including mortality and disease due to higher average temperatures and extreme events (Hunt and Watkiss, 2011).

Excessive heat exposure poses a threat to life, particularly for the most vulnerable in society. Projected increases in the frequency and duration of heatwaves have heightened potential to impact cities due to population density and the Urban Heat Island Effect (UHI) in which waste heat from industry and transport is stored in the materials of the built environment causing urban temperatures to be higher than surrounding rural areas (Jenkins *et al*., 2014).

A key measure of spatial form in cities is the density of development and associated degree of urban sprawl resulting from land use policy. The inverse relationship between residential density and energy consumption from transport suggests that policies which minimise urban sprawl can be used to mitigate against climate change (Larson *et al*., 2012); however, cities with higher densities are faced with more acute adaptation issues such as flooding and urban heat islands with greater intensity (Dawson, 2007).

The demands of climate change mitigation require the transition away from fossil fuels as the primary source of energy; emissions targets will determine government policy leading to a change in urban mobility which is also subject to the impact of escalating fuel prices. Systems are required which can model this transition, combining mitigation policies on transport such as redirected infrastructure investment and incentives to switch fuel types, with carbon taxation and land use. There is a need to incorporate environmental considerations into urban planning to assess potential policy impacts on climate change mitigation and adaptation; the ability to refine policies to address environmental concerns and model the resulting urban system would provide valuable insight (Wegener, 2014).

Urban policies required by goals set out in the Paris Agreement could be explored by linking models of land use and transportation with models of GHG emissions and extreme weather events allowing alternative land use and transport policies to be assessed in terms of both mitigation and adaptation using models of rapid change. By integrating climate hazard models with urban configuration and associated exposure, the areas which are most at risk for a given scenario can be identified, thus the trade-offs between mitigation and adaptation can be explored by policy makers (Ford *et al*., 2018).

1.2 Decision Support

The interaction between policies aimed at mitigation and those aimed at adaptation requires detailed planning to avoid positive interventions aimed at one strategy leading to negative consequences in another (Dawson, 2011). To support decision-making in this domain characterised by conflicting agendas it is necessary to model multiple possible scenarios of future land use and transport corresponding to a suite of planning interventions to evaluate policy implications including environmental impacts; this process should involve multiple stakeholders to inform discussion, seeking compromise to formulate transition plans which are in sympathy with strategies for both mitigation and adaptation.

Improvements to the conceptual design and deployment of models is needed to engage community with the requirements of mitigation and adaptation. Participatory approaches based on integrated modelling have great potential to explore solutions and understand the consequences of policy change. In the context of urban mitigation and adaptation policies, there is a need to convey the benefits and side-effects of proposals to the public. Long-term projections of socio-economic, land use and climate changes are subject to great uncertainty. A range of plausible future scenarios can be used to address this uncertainty to an extent; however, formal analysis of uncertainties and sensitivities is required to handle uncertainties in model inputs, tracking their effects to model outputs (Ford *et al*., 2018).

Systems are required which consider planning options at a range of spatial and temporal scales, which take climate hazards into account as policy constraints or impact assessments for different planning options, and which allow the robust exploration of model uncertainties and sensitivities.

Integrated models of land use and transport generally combine land use, socio-demographic and transport components into model execution, adopting a range of techniques for the computation of each stage which vary in complexity. The current trend is towards models with greater complexity which increases model execution time and adds significant data requirements resulting in difficult model calibration. The need to integrate environmental factors to investigate future policies regarding energy and climate change is an area requiring further research (Acheampong and Silva, 2015).

The dominant approach to urban modelling aims to predict the future state of the urban system with some degree of accuracy using complex models; contrasting this is the use of simpler models whose objective is not to predict, but to explore the parameters and scenario space across multiple model iterations to inform discussion (Batty, 2013).

There is a scarcity of simple urban models which are suited to the task of integrating land use and transportation with the environment; in many cases the data requirements, difficulty of calibration and slow execution times prevent the adoption of existing models for this purpose (Mikovits *et al*., 2014).

SIMULACRA is a framework to vary and rapidly generate multiple instances of a generic spatial interaction model to generate zonally disaggregate values. The model incorporates transport accessibility and handles four sectors; employment, population, retail and industry which can be combined in any order allowing constraints to be resolved flexibly across zones and sectors to balance the system using iteration (Batty *et al*., 2013).

In (Mikovits *et al*., 2017) the authors describe the development of a simple urban model based on random utility which uses limited input data to generate land use scenarios at the level of vector-parcels representing development blocks suitable for the investigation of urban drainage and flooding. The emphasis is on performance, ease of use and integration with models of urban water systems such as a hydrodynamic sewer model; the application of the model generates a range of stochastic development patterns linked to urban drainage and flooding which reduces the required detail and accuracy of the land use model.

SLEUTH is a simple and extensively used cellular automata model which uses transition rules to simulate urban growth and land use change at the level of raster cells. It makes use of six raster inputs; **S**lope, **L**and cover, **E**xcluded, **U**rban, **T**ransportation, **H**ill-shade from which its name is derived. Five input coefficients control growth rate in terms of; dispersion, breed, spread, slope resistance and road gravity. These coefficients are used to adjust the influence of four cellular growth factors; diffusive, organic, new spreading centre and road influenced.

SLEUTH can be used in scenario modelling for policy testing by making use of the exclusion layer and has been coupled with other models to explore environmental issues such as hydrology (Chaudhuri and Clarke, 2013).

The three simple models described above each have desirable properties in terms of exploring scenarios of land use alongside climate change mitigation and adaptation, however, none of them possess the complete set of model requirements; SIMULACRA cannot produce outputs at the fine-scale required for interaction with most sources of environmental data; the model described in Mikovits *et al* (2014) does not account for transport or planning policy; finally, SLEUTH is perhaps too simple to capture detailed land use and transport policies and model calibration is computationally expensive.

In terms of the complexity of urban land use and transport models, each planning problem has bespoke model requirements regarding scale and complexity when the practical tasks of data gathering, calibration and model execution time are considered; the optimum model is one which outputs just enough detail to address the problem whilst minimising these practical costs. “Future urban models will be modular and multi-level in scope, space and time.” (Wegener, 2011, pp 171)

Integrated land use, transport and environment models are rare; one example is the Urban Integrated Assessment Framework (UIAF) developed by the Tyndall Centre for Climate Change Research (Hall *et al*., 2009) which models urban processes at multiple scales to assess the interaction between policies, climate impacts and emissions; the framework has been successfully applied to the Greater London Authority area in a stakeholder dialogue setting. Further research is needed to develop an operational generic decision support tool capable of model exploration for a range of planning problems in multiple settings.

1.3 Aims and Objectives

The aim of this research is to develop a flexible modelling framework to provide decision support to urban planners and stakeholders engaged in participatory modelling; by exploring the tensions and trade-offs of alternative spatial planning policy in scenarios of future land use and transportation, urban transitions can be made which account for climate change mitigation and adaptation measures.

The thesis has 5 distinct objectives to meet this aim:

Objective 1: Explore the link between spatial planning and urban form in the context of climate change and sustainability to identify key drivers of spatial planning policy

Objective 2: Specify the modelling requirements to best support the decision-making processes identified in objective 1.

Objective 3: Review the field of urban modelling to identify and assess candidate modelling approaches.

Objective 4: Develop a modelling framework using techniques identified by objective 3, to provide decision support for planners and meet the requirements specified in objective 2.

Objective 5: Apply the modelling framework to study regions and modelling scenarios to demonstrate the utility of the approach.

1.4 Thesis Structure

The remainder of this thesis responds to the aims and objectives presented in Section 1.3 and is composed of six chapters. In chapter 2 the field of urban planning is reviewed to identify challenges faced by policy makers and decision support requirements in the context of climate change and sustainability. Following this, the theoretical background of key urban modelling trends is provided along with a critical analysis of their approach. This analysis is then used in conjunction with the decision support requirements to formulate an appropriate modelling approach. Chapter 3 outlines the methodology of a multi-scale flexible framework for urban modelling while chapters 4-6 describe its software implementation in detail. In Chapter 7 the framework is applied to a case study for London, a large urban area, using multiple modes of transportation and detailed spatial planning policy for model calibration. Chapter 8 outlines a case study for the Alpine city of Innsbruck and the state of Tyrol in western Austria in which focus is placed upon model calibration using a minimum set of input data. Chapter 9 critically assesses the utility of the developed framework, discussing the London and Innsbruck case studies and a range of further model applications. The final chapter concludes the thesis by summarising the research findings with respect to the aims and objectives set out in Section 1.3 and identifies several avenues of future research.

Chapter 2. Literature Review

2.1 Introduction

This chapter establishes the research gap to be addressed by this thesis; section 2.2 examines policy and spatial planning in the context of sustainable development including environmental considerations and establishes the need for advanced decision support tools to better assess the sustainability impacts of development; section 2.3 reviews urban modelling theory and identifies techniques to develop an appropriate modelling and assessment system.

2.2 Sustainable Urban Development

The concept of sustainable development came to prominence after the publication of the United Nation's World Commission on Environment and Development (Brundtland, 1987) which detailed future challenges faced by urban areas; in response, governments have adopted the concept and dedicated independent institutions have been formed. The Brundtland Report gives a somewhat vague definition of sustainable development as follows: "…meeting the needs of the present without compromising the ability of future generations to meet their own needs." (UN, 1987)

A more specific definition is based on the work of Pearce et al. (1989) which defines a sustainable development framework as comprising economic growth, social justice and environmental protection; often referred to as the 'three pillars of sustainability' (Gibson, 2006). The economic role is focussed on supporting economic growth through the provision of land and infrastructure; the social role is focussed on the provision of housing and services in support of health, social and cultural wellbeing; while, the environmental role is focussed on the protection and enhancement of the environment, including biodiversity and natural resource management. The definition is typically interpreted as a Venn diagram where sustainable development is defined by the intersection of three rings representing the environment, the economy and society. The UK government adopts this 'three-pillar' definition in the 'Sustainable Development Strategy' (Defra, 2005) and the 'National Planning Policy Framework' (DCLG, 2011).

Sustainability efforts are concentrated on high density populations in urban areas and the continuing process of urbanisation. In turn, urban sustainability is increasingly considered in the context of climate change (Carter, 2011). The majority of the world's population now live in urban areas and the trend towards urbanisation is set to continue; global population growth projections indicate that between 2000 and 2050 the capacity of urban areas will need to be doubled in the case of developed countries and increased by over 300% in developing nations (UN, 2012).

This global urbanisation drives the expansion of existing urban areas, increasing the demands on land use, resources and infrastructure. Cities account for around 80% of global economic function and roughly 75% of global resource consumption, comprising energy and material flows. The current reliance on mostly finite resources to fuel economic growth places the process of urbanisation at the forefront of sustainability research. This focusses on decoupling economic growth from the increased use of resources, and reconfiguring urban infrastructure to provide more efficient intra-urban resource flows (Hodson *et al*., 2012).

The Paris Agreement (UNFCC, 2015), commits signatories to limit anthropogenic climate change requiring a rapid reduction in GHG emissions to slow the global rate of temperature increase. This mitigation effort must be coordinated with adaptation challenges posed by increasingly frequent and severe weather events accompanying existing climate change. Further to the Paris Agreement signed by national governments, city-level measures to reduce GHG emissions and adapt to climate change have been agreed by many of the world's city leaders (UCLG, 2015). Most of the world's urban areas with population greater than 5M inhabitants are situated within 100 km of the coast (Nicholls *et al*., 2007); this near-coastal region is estimated to house around a quarter of the world's population at a density nearly three times the global average (Small and Nicholls, 2003). The spatial expansion and densification of these areas increases exposure to the impacts of climate change; projected rises in sea level will lead to increased flooding and submergence in coastal zones. Poverty increases vulnerability to climate-related hazards which exacerbates other problems resulting from uneven development processes. By limiting the magnitude of climate change the overall risks of severe and irreversible impacts can be reduced (IPCC, 2014).

Cities consume resources and generate greenhouse gas emissions disproportionately to their spatial extent, they are also especially vulnerable to disrupted resource supplies and the effects of climate change (McEvoy *et al*., 2012).

The most likely critical effects of climate change on cities include:

- Impacts on energy consumption for heating and cooling.
- Impacts on water availability.
- Impacts of rising sea levels and storm surges on coastal areas.
- Impacts on built infrastructure of extreme events such as storms, flooding, heat extremes and droughts.

- Impacts on health including mortality and disease due to higher average temperatures and extreme events (Hunt and Watkiss, 2011).

The interaction between policies aimed at mitigation and those aimed at adaptation requires detailed planning to avoid positive interventions aimed at one strategy leading to negative consequences in another (Dawson, 2011); table 2.1 gives examples of some of these policy trade-offs.

Response	**Potential benefit**	**Potential negative impact**
Air conditioning	Reduce heat stress	Increase energy needs and GHG emissions
Densification of cities	Reduce GHG emissions	Increase urban heat island intensity and noise pollution
Desalination plants	Secure water supply	Increase GHG emissions
Irrigation	Supplying water for food	Salinization of soil and degradation of wetlands
Biofuels for transport and energy	Reduce GHG emissions	Deforestation, replacement of food crops, raised food prices, air quality pollutants
Catalytic convertors	Improve air quality	Large scale mining and resource movements
Cavity wall insulation	Reduce GHG emissions	Increased flood damage
Raise flood defences	Reduce flood frequency	Encourage more development
Pesticides	Control vector borne disease	Impact on human health, increased insect resistance
Conservation areas	Preserve biodiversity and ecosystems	Loss of community livelihoods
Insurance or disaster relief schemes	Spread the risk form high-impact events	Reduce longer-term incentive to adapt
Traffic bypasses or radial routes	Displaces traffic emissions from city centre, improving air quality and reducing noise	Can increase congestion and journey times (and consequently GHG emissions)

Vehicle user charging	Discourage vehicle use to reduce GHG emissions	Lead to greater social inequality

Table 2.1: Example trade-offs between mitigation, adaptation and sustainability policies (Dawson, 2011)

There is considerable debate as to what constitutes sustainable urban form with conflicting ideas about which sustainability measures are most important and how they should be addressed.

The Compact City model was conceptualised by Dantzig and Saaty (1973) to constrain urban sprawl and hence protect surrounding rural areas. Advocates of the model claim that more dense cities exhibit a reduced reliance on private vehicles which in turn reduces expenditure on fuel, GHG emissions and places a greater emphasis on the provision of public transport. In terms of policy, the model has had a significant impact on national planning policy including in the UK and is implemented by development constraints in greenbelt land and the promotion of brownfield sites for redevelopment (Williams, 2004). Critics dispute the central claim that reliance on private vehicles is reduced and point to the potential implications of increased congestion, poorer air quality and the exacerbation of the urban heat island effect.

The review of urban intensification policies in Melia et al. (2011) demonstrates that the implementation of the Compact City model resulted in higher congestion and poorer air quality. Transit Oriented Development (TOD) is a conceptual response to the unsuitability of some urban areas for densification and reflects the need for change in residents' commuting habits. The model promotes a more polycentric view of the city and emphasises development and the provision of services around transport hubs which are linked to other centres. In terms of policy, the model is partly implemented by improved transport infrastructure and increased land use densities around major transport stations (Cervero et al., 2002). Advocates of the model point to lower emissions due to sustainable transport policies whereas critics suggest that reliance on private vehicles for transport is not significantly reduced when compared with compact cities.

The concept of Garden Cities was originally developed by Howard (1902) and has been revisited in the context of sustainable development. The model emphasises the proximity between the urban population and nature by using multiple development centres, each of which is surrounded by a greenbelt. The historic dominance of the Compact City model in planning has minimalised the implementation of the model but it has been revived by recent research into sustainable urban form (Randolph, 2013). In relation to the Compact City model and TOD: The peripheral development in the Green City concept can alleviate development pressures resulting from densification attributed to the Compact City model; whilst the aim of reducing private vehicle use and emissions by developing around public transport stations and along existing routes is in sympathy with TOD. Opposition to the model is in the form of reluctance to develop satellite urban areas in existing greenbelt land and concerns about increased emissions due to longer journeys.

The UK employs the use of sustainability appraisals (DCLC, 2011), stipulated as part of the National Planning Policy Framework (DCLG, 2011) during the preparation of a local development plan (ODPM, 2004); local requirements form the basis for specific sustainability plans which are used to form objectives for the sustainability appraisal of proposed development. The appraisal is based on economic, social and environmental considerations and expects proposals for new development to seek opportunities to provide net gains across all three categories. The proposed development options are assessed against sustainability criteria to compare alternatives and identify the most sustainable options. The consideration of climate change impacts and conflicting policies between development alternatives is highly limited and whilst the appraisals are broad in scope, they are subjective and lacking in quantitative detail. This element of the planning process has attracted criticism where a more detailed and analytical treatment of evidence and policy interaction is needed (Gibson, 2006). In addition, whilst the UK planning process requires the modelling of transport, there is no equivalent requirement for land use modelling to assess potential impacts. This lack of detail in the implementation seems to be contrary to the supposed shift in UK spatial planning from the use of sustainability as one factor in the consideration of competing land uses, to the stated principle aim of the planning system (DCLG, 2011).

Decision support tools involving sophisticated spatial modelling could be integrated into the planning process to address these concerns (Geertman and Stillwell, 2009). Hansen (1959) demonstrated that accessibility and land availability can be used to model residential land use. Since then a vast array of models have been developed which model the spatiotemporal evolution of land use and transportation, in recent times emphasis has been placed upon linking these with environmental models to assess the impact of urban development. Decision support tools which allow detailed analysis have potential to inform and positively steer the planning process (Gasparatos et al., 2008).

To support decision-making in this domain characterised by conflicting agendas it is necessary to model multiple possible scenarios of future land use and transport corresponding to a suite of planning interventions to evaluate policy implications including environmental impacts; this process should involve multiple stakeholders to inform discussion, seeking compromise to formulate transition plans which are in sympathy with strategies for both mitigation and adaptation.

The demands of climate change mitigation require the transition away from fossil fuels as the primary source of energy; emissions targets will determine government policy leading to a change in urban mobility which is also subject to the impact of escalating fuel prices. Systems are required which can model this transition, combining mitigation policies on transport such as redirected infrastructure investment and incentives to switch fuel types, with carbon taxation and land use. There is a need to incorporate environmental considerations into urban planning to assess potential policy impacts on climate change mitigation and adaptation; the ability to refine policies to address environmental concerns and model the resulting urban system would provide valuable insight (Wegener, 2014).

Improvements to the conceptual design and deployment of models is needed to engage community with the requirements of mitigation and adaptation. Participatory approaches based on integrated modelling have great potential to explore solutions and understand the consequences of policy change. In the context of urban mitigation and adaptation policies, there is a need to convey the benefits and side-effects of proposals to the public (Ford et al., 2018).

2.3 Urban Modelling

In an empirical study of Washington, DC, Hansen (1959) demonstrated that accessibility and land availability can be used to model residential land use; the potential for growth and the density of that growth is proportional to the accessibility of the location. This co-dependent link between land use and transportation paved the way for coordinated planning and formed the basis for extensive further research.

In his *Model of Metropolis*, Lowry (1964) developed the first operational model which integrates land use and transport using nested spatial-interaction based on the law of gravitation to model residential and service/retail employment locations within a zonal system. The model incorporates transport in the form of trips between residences and workplaces, trips from residences to retail and the travel time between zones. Linked by assumptions on the balance of residents and workers via ratios of activity and inverse activity, the two models are iterated until the system reaches equilibrium generating the final zonal values of employment and population. A more robust theoretical framework for spatial-interaction location is grounded in the theory of entropy maximisation, where entropy refers to the degree of system disorder i.e. the location of residents and workers (Wilson, 1970). Four types of spatial-interaction are defined: unconstrained, in which the location of residences and workplaces is not fixed; production-constrained, in which households seek workplaces; attraction-constrained, in which households seek residences; and doubly-constrained, in which the locations of residences and workplaces is fixed.

The economic basis for land use assumes that accessibility, attractiveness and market value are directly proportional when choosing between development locations. In his description of *bid rent,* Alonso (1964) formalises the urban economy in a theory where the locational choice of households and firms is determined by matching their *bid rent* to the asking price of the land owner resulting in equilibrium of the land market. Firms generating a higher monetary value per unit of land can make higher bids and hence occupy more attractive locations with higher accessibility. Accessibility-based location models build upon Alonso's theory of *bid rent* using accessibility indicators of varying complexity to predict the opportunity for spatial-interaction at potential locations. A commonly used complex indicator of accessibility combines all destinations of interest with an inverse function of the cost of reaching them in terms of time, money or both i.e. generalised cost (Wegener, 2014).

Random Utility Theory (RUT) developed by McFadden (1973) provides a framework for modelling the complex behavioural dynamics of individual choice when applied to decisions of locational accessibility, utility and migration in land use and transport modelling. Choice behaviour sampled from a representative population is used to statistically adjust the model of individual choice to reflect individual choices which are based on preference but also exhibit random variation. The link between individual choice behaviour and the distribution of decision rules in the population allows the representation of choice when faced with a set of alternatives. The theories of Lowry (1964), Wilson (1970) and McFadden (1973) were combined to embed discrete choice theory into a model of the urban economy, introducing several factors pertaining to choices and preferences (Anas, 1984). Most operational accessibility-based location models build upon this work by applying discrete choice models to functions of utility combining accessibility with various measures determining the attractiveness of potential locations.

Action-Space or Time-Geography theory (Hagerstrand, 1970; Chapin 1974) predicts the patterns of land use resulting from activities attributed to different population groups using "time budgets" to distinguish between groups' mobility, income and social role. Different groups are assigned different action spaces corresponding to available opportunities which are subject to three constraints: capacity, the ability to overcome space in time; coupling, the need to coordinate time with others; and institutional, access restrictions at locations. These restrict spatial-interaction for each group according to their “time budgets” where the generalised cost of performing a set of activities is minimised.

General theories on system dynamics such as those developed by Forrester (1969) have been applied to urban areas using a representation of the urban system comprising three categories for businesses, housing and people respectively. Factors such as land density and the availability of employment are represented as “rates of flow” which flow into and out of categories. The numbers present in each category for the current interval is determined by flows into and out of all categories from all previous intervals. The feedback across all flows results in a complex system which moves from its initial conditions to equilibrium via a series of oscillations revealing the unexpected short-term consequences of long-term policies.

Using general systems theory, cities can be viewed as complex adaptive systems which are constantly in a state of disequilibrium (Batty, 2007). Long term dynamical system behaviour is determined by initial conditions; however, such systems are subject to feedbacks and can be in one of three states at any one time: chaos, where no rules apply; stability, where behaviour is linear or differentiable; or complexity, where both chaos and stability apply in different periods or regions.

The two simplest approaches to modelling the behaviour of complex systems use Cellular Automata (CA) and Agent-Based Models (ABM); both methods employ agents and actions at the microscale which are extended over time and space producing aggregate behaviour. CA are mapped to a (generally two-dimensional) raster grid of cells and use local interactions to apply a provided set of rules to each cell in turn; this produces a changed grid which is swapped with the initial grid in the next iteration hence extending the application of rules over time and space (Clarke, 2014).

The four key components of a CA are as follows:

1. A grid of cells whose state is one of a finite set;
2. A neighbourhood of cells over which defined actions are applied;
3. A set of initial conditions which initiates the specific state of each cell;
4. A set of one or more rules which change a cell’s state based upon the states of cells within its neighbourhood.

CA can be used to simplify and model complex dynamical systems including feedbacks such as cities, demonstrating emergent behaviour using a minimum set of states, initial conditions and rules (Batty, 2000); this work explored variations in the use of CA including the relaxation of neighbourhood assumptions to incorporate action-at-a-distance and the relationship to simple Cell Space models which apply cell transitions over time. Sante et al. (2010) compared 33 urban CA models and describe further modifications made to strict CA when applied to modelling urban growth including the use of nonuniform cell spaces, irregular timesteps and complex transition rules.

ABMs simulate the behaviour, actions and interactions of autonomous agents representing individuals or collectives. Agents are goal-oriented independent units generally used to explore behavioural impacts and the resulting variations in the system being modelled; in simple cases a single agent can be used for this purpose while more complex cases involving agent-agent interactions can be modelled using a multiagent model. Agents react to system properties and other agents, using this information to "learn" and subsequently adapt their behaviour to better meet their goals (Clarke, 2014).

The five key components of an ABM are as follows:

1. Agents specified in terms of their type and model scale;
2. Heuristics for agent decision-making;
3. Rules which govern how agents learn and adapt;
4. Agent engagement procedure for actions and interactions;
5. An environment which can exert influence on and be impacted by the behaviour of agents.

A key example of ABM as applied to urban systems is the work of Schelling (1971) on residential segregation which conceptualised the use of autonomous agents interacting within their environment to produce an observed aggregate result.

In McDonnell and Zellner (2011) the authors describe the use of a prototype ABM to investigate the primary and secondary impacts of the introduction of Bus Rapid Transit (BRT) involving exclusive lanes and other priority measures. The primary impact is the potential modal shift from cars and reduced congestion in response to quicker bus journeys; the potential secondary impact is the reversion to travel by car which could result from reduced congestion. ABM allows the examination of potential impact interactions along with the travel times and modal share in the system as agents respond to BRT policy incentives.

Another approach to modelling at the individual level is the use of microsimulation to generate synthetic populations of agents whose behaviour is sampled from more aggregate data. This method is adopted by Waddell et al (2003) in the design of the UrbanSim system which represents the most detailed integrated model of land use and transport in terms of disaggregation and dynamics. In its use of microsimulation, UrbanSim generates synthetic populations from census data and operates at the level of individuals and households; every person, household, building, job etc is represented and models of individual choice are used extensively drawing upon random utility maximisation and discrete choice theory. In recent work (Waddell, 2018), an ABM of travel demand, ActivitySim is being developed which integrates with UrbanSim to model the travel choices of synthetic agents.

The overwhelming trend in urban modelling is the disaggregation towards microsimulation; this is driven by the development of more powerful computers and the availability of disaggregate data from GIS-based applications. These resources are coupled with complex systems approaches using CA and ABM, and models of both land use and transport using microsimulation. Modelling at the individual level has the benefit of improved conceptual theories of behaviour, interactions and preferences which influence mobility and patterns of location; however, this must be balanced against the associated practical drawbacks of large data requirements and long computing times. Microsimulation models are also prone to a lack of stability arising from stochastic variation when different random seeds are used across model iterations; this can obscure the response to variations of model inputs preventing spatial policy assessment (Wegener, 2014).

The dominant approach to urban modelling aims to predict the future state of the urban system with some degree of accuracy using complex models; contrasting this is the use of simpler models whose objective is not to predict, but to explore the parameters and scenario space across multiple model iterations to inform discussion (Batty, 2013). There is a scarcity of simple urban models which are suited to the task of integrating land use and transportation with the environment; in many cases the data requirements, difficulty of calibration and slow execution times prevent the adoption of existing models for this purpose (Mikovits *et al*., 2014).

Long-term projections of socio-economic, land use and climate changes are subject to great uncertainty. A range of plausible future scenarios can be used to address this uncertainty to an extent; however, formal analysis of uncertainties and sensitivities is required to handle uncertainties in model inputs, tracking their effects to model outputs (Ford *et al*., 2018).

Spatial-interaction location models have the positive attribute of being rooted in the economic theory of production and consumption; however, the treatment of households as industries within this system implies that workplace location is the sole determining factor for the choice of household location. Accessibility-based location models improve on this by their ability to consider a range of different accessibility indicators to reflect different needs; this introduces system lag since indicators are computed for each timestep to inform the generalised cost of travel, and hence choice of location in the following timestep. Accessibility-based location models also have the advantage of separating land use and transport model components allowing the development of bespoke sub-models for location behaviour; this modular structure has positive implications in terms of the software implementation of the model which contrasts with the approach of spatial-interaction location models which are unified by a single complex equation (Wegener, 2014).

Integrated models of land use and transport generally combine land use, socio-demographic and transport components into model execution, adopting a range of techniques for the computation of each stage which vary in complexity. The current trend is towards models with greater complexity which increases model execution time and adds significant data requirements resulting in difficult model calibration. The need to integrate environmental factors to investigate future policies regarding energy and climate change is an area requiring further research (Acheampong and Silva, 2015).

In terms of the complexity of urban land use and transport models, each planning problem has bespoke model requirements regarding scale and complexity when the practical tasks of data gathering, calibration and model execution time are considered; the optimum model is one which outputs just enough detail to address the problem whilst minimising these practical costs. “Future urban models will be modular and multi-level in scope, space and time.” (Wegener, 2011, pp 171)

Integrated land use, transport and environment models are rare; one example is the Urban Integrated Assessment Framework developed by the Tyndall Centre for Climate Change Research (Hall *et al*., 2009) which models urban processes at multiple scales to assess the interaction between policies, climate impacts and emissions; the framework has been successfully applied to the Greater London Authority area in a stakeholder dialogue setting. Further research is needed to develop an operational generic decision support tool capable of model exploration for a range of planning problems in multiple settings.

2.4 Summary

This chapter has established the research gap to be addressed by this thesis; section 2.2 examined policy and spatial planning in the context of sustainable development including environmental considerations and established the need for advanced decision support tools to better assess the sustainability impacts of development; section 2.3 reviewed urban modelling theory and identified techniques to develop an appropriate modelling and assessment system. The following chapter presents the methodological framework for this thesis and begins with a detailed discussion of the model requirements for decision support.

Chapter 3. Methodological Framework

3.1 Introduction

Building on the review of the literature in Chapter 2 this chapter describes the methodology of an integrated urban modelling system for decision support in the context of climate change mitigation and adaptation planning. The chapter first describes the need for an integrated approach to modelling future land use, transport and the environment (Section 3.2), before defining desirable characteristics required by such a system in Section 3.3. It then reviews existing urban modelling frameworks and considers their relative strengths and limitations (Section 3.4). Section 3.5 discusses the concept and models of the Urban Integrated Assessment Framework (UIAF) before critiquing the framework in terms of requirements identified in Section 3.3. In Section 3.6 the conceptual design and implementation of an Urban Integrated Modelling Framework (UIMF) is presented which reworks and integrates existing models in order to address problems identified in Section 3.5.

3.2 Problem Formulation

The Paris Agreement of December 2015 commits signatories to limit anthropogenic climate change requiring a rapid reduction in greenhouse gas (GHG) emissions to slow the global rate of temperature increase (UNFCC, 2015). The challenges of mitigation must be addressed alongside those of adaptation posed by increasingly frequent and severe weather events accompanying existing climate change. Densely populated urban areas are spatially disproportionate contributors to global emissions and are at an increased risk from climate change and extreme weather events (Walsh *et al*., 2011). In response to this disproportionality, and further to the Paris Agreement signed by national governments, city-level measures to reduce GHG emissions and adapt to climate change have been agreed by many of the world's city leaders (UCLG, 2015, Reckien *et al*., 2018).

The timeframe set out in the Paris Agreement requires that rapid change is implemented if signatories are to fulfil their obligations; by exploring scenarios of radical change in urban systems and gaining an understanding of possible consequences, cities could pave the way to global change as required by the Paris Agreement (Bai *et al*., 2018). City-scale policies aimed at climate change mitigation and adaptation should lead to an improved quality of life for inhabitants; the sharing of knowledge and experience in tackling these challenges between different urban areas and in different countries could spread this improvement globally.

Strategies to drastically reduce GHG emissions (Rockstrom, 2017) will dictate national and local policies on both the production and consumption of energy and will have impacts on urban mobility. Many cities have formulated plans to mitigate against further climate change, but adaptation measures are less well addressed (Heidrich *et al*., 2016). Extremes of higher temperatures, more intense rainfall and water scarcity will impact both human health by increasing morbidity and mortality (Harlan and Ruddell, 2011), and urban systems locally and remotely via disruption to interdependent resource networks (Pregnolato *et al*., 2016). Concurrently with the commitment to mitigation, improved measures for risk adaptation associated with observed climate change must be developed to improve infrastructure resilience and protect urban inhabitants.

Adaptation-targeted policies for land use and transportation aim to reduce the risk to urban areas by limiting the vulnerability of inhabitants and their exposure to climate hazards (Funfgeld, 2010). Urban policy must be informed by detailed consideration of the possible consequences of planning decisions including land use and transportation. Tensions between policies aimed at mitigation and those aimed at adaptation require comprehensive strategies to achieve a balanced outcome (Dawson, 2011). By integrating climate hazard models with spatial configuration of urban areas and associated exposure, areas most at risk for a given scenario may be identified, and the trade-offs between mitigation and adaptation can be explored by policy makers.

Long-term changes in policy, population and prosperity drive changes in land use and transportation. Developments in tools linking LUTMs with environmental models could help to manage these transitions and the policies required to steer them (Wegener, 2014). Socio-economic factors could then be linked with environmental factors in scenarios for policy testing using integrated LUTMs and goals set out in the Paris Agreement could be explored by linking LUTMs with models of GHG emissions and extreme weather events allowing alternative land use and transport policies to be assessed in terms of both mitigation and adaptation using models of rapid change.

Improvements to the conceptual design and deployment of models is needed to engage policymakers and the public with the requirements of mitigation and adaptation; models should be used to inform the planning process taking into account a range of considerations and involving multiple stakeholders, demonstrating the feasibility and possible implications of various measures under test.

Models are often used to forecast based on current trends, testing measures aimed at improving transport and boosting economic growth (Batty, 2013a). In the context of climate change mitigation and adaptation, multiple scenarios using relatively simple models could form the basis for further exploration using more detailed approaches where stakeholder dialogue and collaboration is required in a coordinated effort to explore and design for future urban transitions. Long-term projections of socio-economic changes (e.g., employment and population), land use and climate change are subject to great uncertainty (Waddell, 2011). Formal analysis of uncertainties and sensitivities is required to handle uncertainties in model inputs and track their effects to model outputs.

There is a need to convey the benefits and side-effects of proposals to the public. The use of LUTMs in governance should migrate from attempted prediction towards exploration, learning and policy adaptation in a participatory modelling process where public involvement could improve understanding and acceptance, allowing a greater range of more radical policies to be explored. Analysing suites of alternative options to address climate change via integrated LUTMs is key to exploring the conflict between policies and the trade-offs required. The accessibility of simulations could be improved by using simple interfaces using open source software to manage data, generate scenarios and drive model equations providing the opportunity for greater participation and understanding of model outcomes (Ford *et al*., 2018).

The interrelated processes of climate change and socio-economic change should be considered concurrently within climate impact assessment to assess the future state of the system and the potential effectiveness of planned interventions. The dominant approach to climate impact assessment judges observed socio-economic conditions against predicted future climate scenarios which neglects the socio-economic evolution of the system under test (Berkhout, Hertin and Jordan, 2002). As shown in figure 3.1, climate change and socio-economic change in urban systems are inextricably linked; a change in one aspect inevitably leads to impacts in another.

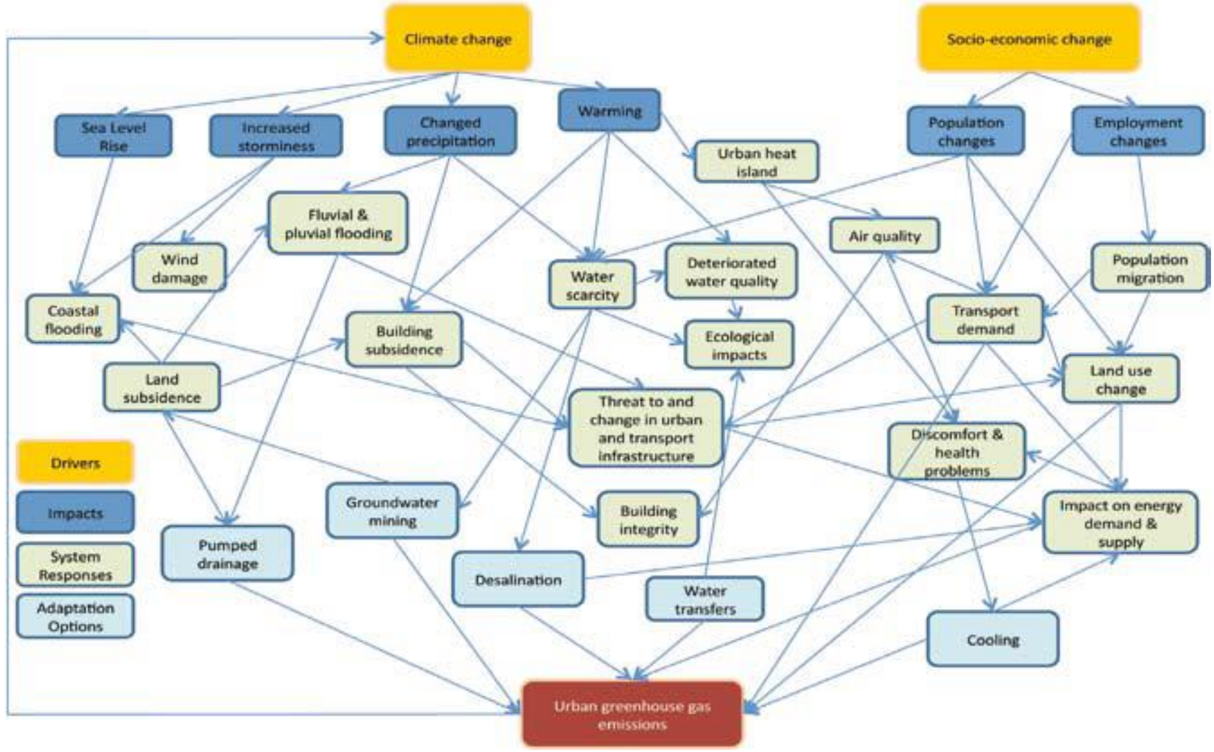

Figure 3.1: Complex Interactions and Interdependencies between Climate Change Mitigation and Adaptation (Hall *et al*, 2009).

The varied timescales of mitigation and adaptation processes coupled with the complexity of their interactions and interdependencies requires a holistic, system-level view to appraise portfolios of policy scenarios across multiple objectives over long-term timescales. A systems approach to cities including interactions of investment, infrastructure, land use and the built environment helps to balance objectives of mitigation and adaptation whilst considering climate change and sustainability alongside economic development (Walsh *et al*, 2011). It is only by the simultaneous consideration of all of these factors and their interactions that the concomitant strategies required to bring about overall benefits whilst minimising adverse effects in any one aspect can be agreed upon.

3.3 Driving Concepts of Integrated Modelling and Simulation

Integrated Assessment and Modelling (IAM) is motivated by the need to handle multiple issues of concern, identifying interdependencies between them and where possible, a solution which satisfies conflicting agendas for the system in question. Taking a holistic view of the system ensures that the interests of all parties are represented and that interventions which modify the function of the system in terms of investment, legislation and control can be assessed in terms of their impact by multiple stakeholders (Hamilton *et al*, 2015). An overview of the key dimensions of IAM from an environmental engineering perspective is shown in figure 3.2. This definition will be used to frame the required capabilities of an integrated modelling framework suited to tackling the problems set out in the previous section. The concepts of key integration drivers, system characteristics and methodological approaches are discussed in the following subsections.

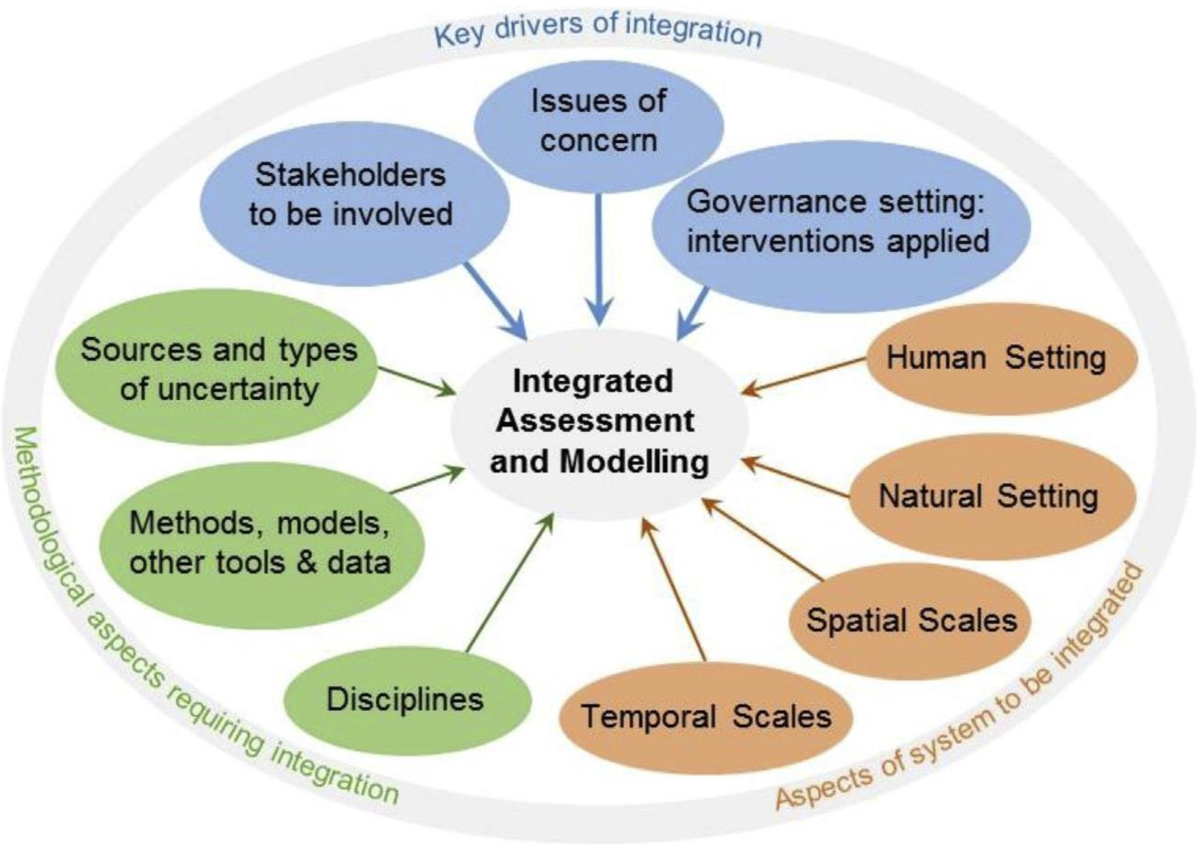

Figure 3.2: Dimensions of Integrated Assessment and Modelling (Hamilton *et al*, 2015).

3.3.1 Key Integration Drivers

The principal issue of concern in the effort to mitigate against climate change is the reduction of GHG emissions. Commitments to limit global average temperature rise below 2 degrees Celsius and ideally below 1.5 degrees Celsius have been agreed between many of the world's leading cities as part of the C40 Cities directive alongside increased resilience to the impacts of existing climate change (C40a).

The global challenge of mitigation can be partly addressed by the collaborative efforts of multiple individual cities whose mitigation pathways differ in accordance with their circumstances. These cities are unified by their commitment to reduce GHG emissions but their approach to achieving agreed targets must be bespoke in order to consider differences in their economies, industries and geographies alongside the needs of inhabitants. In each city, the fundamental issue of concern is how to reduce GHG emissions to the required level within the timeframe agreed. The stakeholders involved include government and politicians who form legislature, industries who adopt new standards and legislation, and individuals who are affected by new laws and associated changes in their living situation. Some member cities of the C40 Cities Climate Leadership Group have already peaked their GHG emissions via a range of mitigation pathways as shown in table 3.1.

C40 Member City	**Year**	**Mitigation Example**
Copenhagen	1991	Decentralisation: expansion of district heating system.
San Francisco	2000	Decarbonisation: switch to renewable energy sources for electricity grid.
Paris	2004	Mobility: infrastructure improvements for public transport and cycling.
Sydney	2007	Buildings: improved energy performance and retrofitting.
Vancouver	2009	Waste: improved composting and landfill gas collection.

Table 3.1: Example Peaked Emission Cities and Mitigations (adapted from C40b).

The plan for Zero carbon London (GLA, 2018) targets a maximum of 1.5 degrees Celsius rise in global temperature and develops strategies for energy including decentralised energy, high electrification and decarbonised gas; strategies for transport including emissions surcharging, ultra-low emissions zones and zero emissions vehicles; and strategies for buildings including zero carbon new buildings and retrofitting for improved energy efficiency. These mitigations each impact on different sectors of industry and the economy and require coordination from national government, the Mayor and Greater London Authority (GLA), London boroughs, businesses and individuals.

3.3.2 System Characteristics

Human settings refer to factors including population, politics and the economy whilst natural settings refer to factors including land use, water and climate. The dependence of urban populations on natural systems for resources, and the impact of human activities on those natural systems leads to tensions between interventions aimed at reducing environmental impacts whilst maintaining perceived quality of life including economic growth and mobility. Measures to mitigate against climate change disproportionately affect poorer communities meaning that the push to tackle climate change must be matched by an effort to reduce inequalities. Legislating the shift towards lower emissions and greater energy efficiency has a greater impact upon lower-earning households who may be unable to meet the costs of conversion without support (C40c). In addition to the costs associated with buildings, the reliance on carbon-emitting vehicles to support employment in many cities is a concern. As well as being more susceptible to negative impacts of mitigation, poorer cities and areas within cities are more vulnerable to the impacts of existing climate change in sectors such as energy, water, waste and public health etc. (C40d). An integrated approach to coupling human and natural systems for both mitigation and adaptation must be developed in order to address these problems concurrently.

The temporal scale of urban processes and interventions is important when seeking to holistically model urban systems; large-scale infrastructure projects such as the construction of new roads can take years to complete and the intermittent disruption may be neglected by models which are based upon snapshots into the future such as relative-static models. In terms of mitigation versus adaptation, global efforts to mitigate against climate change have a long-term payoff versus the short-term payoff of adaptation measures.

Different processes occur at different speeds and at different spatial scales, model outputs are often represented at different spatial scales by area attributes such as zonal land use apportioned to population increase in spatial interaction models, or by uniform grid-cells sampling environmental conditions. This can result in a fundamental mismatch in spatial scale when integrating otherwise valid models which can be referred to as the “Tyranny of Zones” (Spiekermann and Wegener, 1999) in which zonal land use attributed to urban growth in spatially aggregated models cannot be reasonably combined with fine-scale environmental data.

3.3.3 Methodological Approaches

One approach to integrating climate change with urban systems would be to construct a comprehensive model combining all aspects of the system. In theory, a model could be constructed according to a set of principles agreed upon by the modelling team, accounting for all possible conflicts and inconsistencies between modules. However, the level of collaboration required across disciplines to develop such a model, and the complexity of the resulting system favours a more modular approach to integration. When integrating models across disciplines it is inevitable that results will be interpreted differently by groups with different expertise. It must be ensured that the modelling effort is coordinated to ensure that data transactions can take place and that data formats and modelling scales (both spatial and temporal) are agreed. Failure to account for modelling inconsistencies can lead to so called "ugly constructs" in which the coupling of two or more models leads to unexpected results. One approach to model integration is based upon the assemblage of existing models achieved by loose coupling in which the output of one model is fed to the input of another via data files. This reuse and repurposing of existing models has the potential to save time; however, this must be balanced against considerations for model linkage to ensure consistency across independently developed models in terms of parameters and scales (Voinov and Shugart, 2013).

There are many software frameworks which help to bridge the gap between disparate modelling disciplines; a comprehensive review can be found in Granell, Schade and Ostlander (2013). OpenMI can facilitate interoperability between models but can lead to significant overheads in terms of the format and complexity of models developed (Knapen *et al*, 2013).

3.3.4 Framework Requirements

Exploration of how interventions affect systems, or systems-of-systems can be carried out using participatory modelling. IAM is generally conducted via a participatory modelling process as depicted in figure 3.3 which represents modelling stages as cards which can be rearranged flexibly to adapt the process flow in response to feedback between modelling stages. The purpose of modelling in each of these stages can vary as shown in figure 3.4.

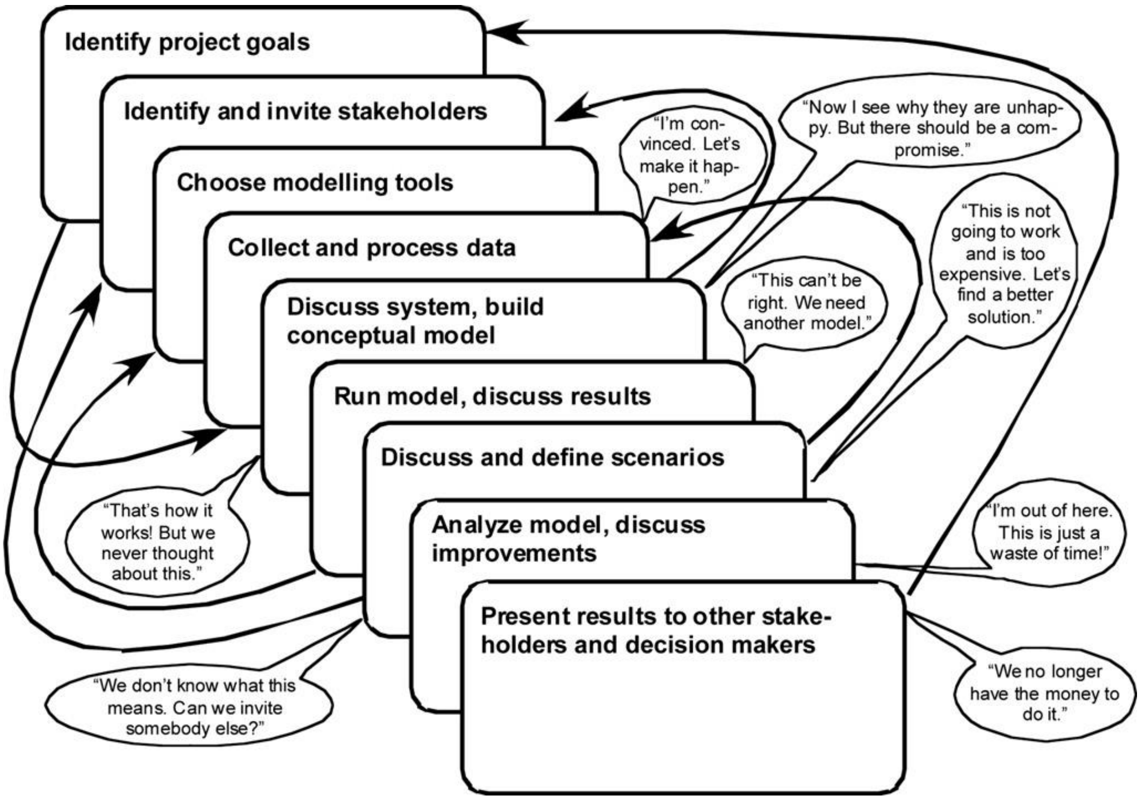

Figure 3.3: Overview of Participatory Modelling Process (Voinov and Bousquet, 2010).

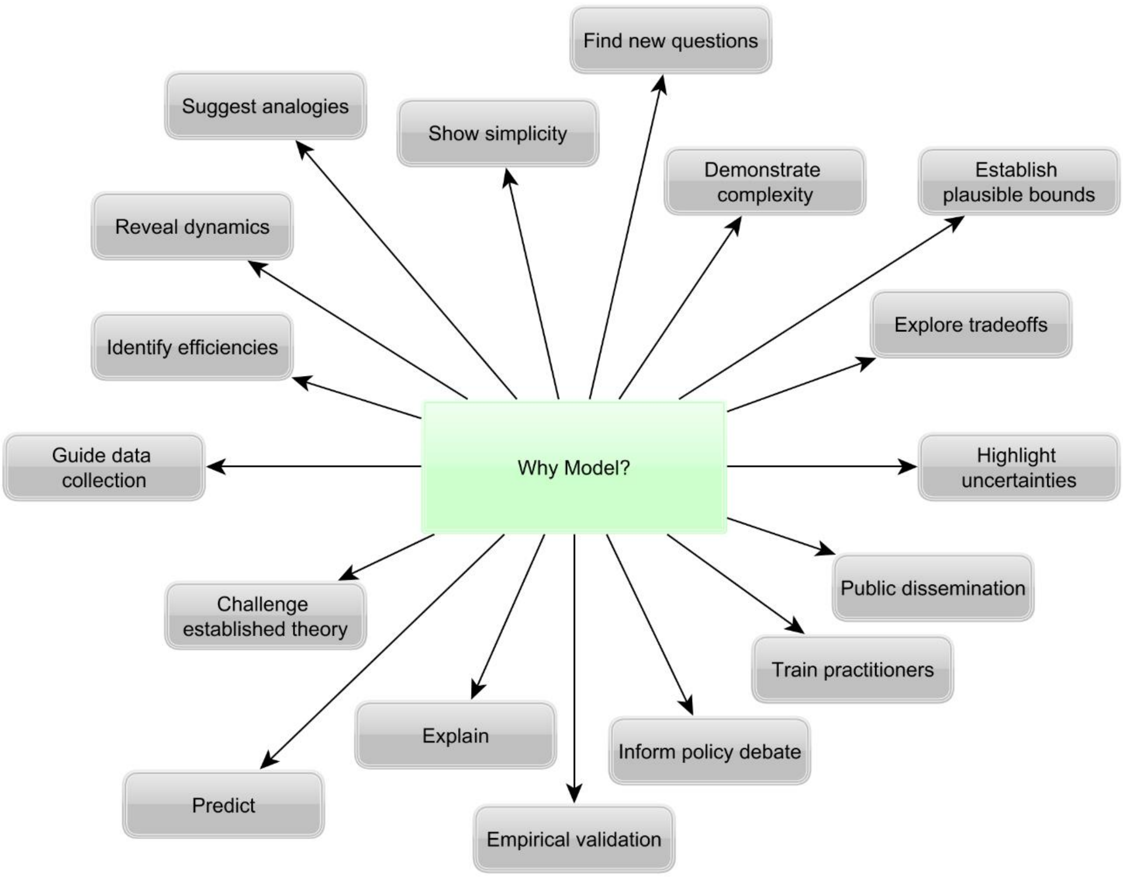

Figure 3.4: Why Model? Adapted from Epstein (2008)

The need to formulate climate change mitigation and adaptation strategies which address the needs of a range of urban inhabitants requires that an integrated approach is taken. For each city these considerations comprise national governance, regional rules, local legislation, businesses and individuals. In order to explore the many problems and potential solutions, an integrated approach should be adopted allowing policy interventions in one sector to be assessed in terms of their impact upon the system as a whole. This approach should be applied to all sectors, culminating in the collation of evidence to inform suites of policy portfolios. Modelling with stakeholders specific to urban systems and climate change leads to the following requirements as shown in table 3.2.

Requirement	**Description**
Validity	The ability of a model to reproduce changes in observed data should be demonstrated in model validation to minimise bias and provide confidence in model outputs
Transparency	Decision support tools aimed at resolving conflicting opinions on urban policy require transparency to be of use; opaque, or 'Black Box' models do not convey sufficient understanding to support argument. Transparency must be balanced with validity for a model to be of practical use i.e. simple models may provide greater transparency at the expense of reduced validity in representing the problem; whereas, complex models may provide a more valid problem representation at the expense of reduced transparency
Usability	To be deployed independently of developers, a model should be able to be configured, adapted and understood by model users and stakeholders. Usability should therefore be maximised whilst maintaining behavioural and empirical validity. Collating suitable data whilst accounting for errors and omissions, and ensuring consistency poses a significant task which is not helped greatly by currently available software. Exhaustive input requirements and the complexity of data development can deter users and prevent model implementation.
Flexibility	To be applicable to a wide range of users and purposes a model system must have enough flexibility to handle variations in input data and model output requirements. The development of decision support tools must also respond to advances in data, software and theory to provide new solutions.

Performance	Computational performance must be considered alongside the other challenges of integrated modelling; for instance, poor performance can restrict the use of otherwise viable models whereas models which aim for interactive runtimes can compromise validity.

Table 3.2: Framework Requirements – adapted from Waddell (2011).

3.4 Existing Urban Modelling Frameworks

Section 2.3 presented an overview of the many existing theoretical approaches to modelling urban systems, each of which has relative benefits and drawbacks. The question is whether they are well-suited to the task of rapid assessment in a participatory modelling process? A suitable approach to modelling land use, transport and environmental factors in an IAM setting must be established in order to design such a system. Table 3.3 provides an overview of existing urban modelling frameworks which are considered and critiqued against the concepts and requirements for modelling with stakeholders described in section 3.3.

Modelling Paradigm	**Example Models and Considerations**
Aggregate spatial interaction	Lowry-Garin model (Lowry, 1964) MEPLAN (Echinique *et al*, 1990) TRANUS (de la Barra, 1989) **Pros:** Rooted in economic theory of production and consumption **Cons:** Implies that workplace location is sole determinant for household location
Aggregate utility	DELTA (Simmonds, 1999) IRPUD (Wegener, 2011a) Uplan (Walker, Gao and Johnston, 2007) **Pros:** Choice of location considers a range of needs via accessibility indicators - Separation of land use and transport allows modular development of bespoke sub-models of location behaviour **Cons:** Accessibility indicators for each timestep inform location in the next timestep introducing system lag - Greater data requirements and longer computing times

Micro-simulation	ILUMASS (Moeckel, Schumann and Wegener, 2003) ILUTE (Salvini and Miller, 2005) UrbanSim (Waddell *et al*, 2003) **Pros:** Improved validity of behaviours and interactions using agent-based modelling - Greater spatial disaggregation **Cons:** Significantly greater data requirements and much longer computing times - Lack of stability due to stochastic variation
Cellular Automata	SLEUTH (Silva and Clarke, 2002) **Pros:** Relatively simple and flexible approach to modelling urban growth based on past development **Cons:** Difficult to incorporate top-down policy drivers – long model execution times

Table 3.3: Existing Urban Modelling Frameworks

3.4.1 Aggregate Spatial Interaction Models

Aggregate spatial interaction models simulate flows of activities such as traffic or household relocation between geographical zones in an urban region. These origins and destinations of flows are typically represented by point locations at the centroid of each zone. Interactions between locations are in proportion to the product of masses at origins and destinations, and inversely proportional to a measure of their separation. This gravity model is adapted from Newton's second law of motion to consider the measure of separation between origins and destinations in terms of the distance, time or cost of travel. The use of spatial interaction is well established in the four-step transport model (trip generation, trip distribution, modal split, traffic assignment) where trip distribution is based upon a gravity model. This distribution can be used to predict activities at particular locations using the concept of potential which in this case is defined as the sum of interactions between one origin and all destinations or vice versa (Batty, 2007a).

A key characteristic of spatial interaction models is that households are represented as industries which produce labour and consume commodities. Household location is then predicted using a multi-industry input-output model based upon production and consumption. This provides model foundations in economic theory but implies that household location is based solely upon workplace location, neglecting other aspects of residential choice.

3.4.2 Aggregate Utility Models

The simple relationship between the location of workplaces and households used in spatial interaction is expanded upon by aggregate utility models to represent a wider range of locational choice factors. These models are based upon accessibility which is defined as the opportunity for spatial interactions at any given location. Bespoke accessibility indicators can be developed to predict the location of workplaces and households using discrete choice models which consider a range of activities. The potential accessibility of any given location can be found by summing activities across destinations and weighting by an inverse function of the distance, time, or generalised cost of reaching them. Potential accessibility can then be combined with other zonal attributes to determine the attractiveness of locations in terms of each type of workplace and household thus representing a range of different needs (Wegener, 2014).

The use of bespoke zonal attractors based on accessibility indicators allows the location choices made by different types of workplace and household to be simulated. This modelling requires more data than spatial interaction but better represents the needs of different groups within the system. A key implication of using accessibility is that the resulting transport demand can only be known in the following timestep.

3.4.3 Microsimulation Models

The disaggregation of locational choice by workplace and household type as used in utility models is taken further in microsimulation models which represent the urban system as an aggregation of individual agents and interactions. These agents can represent the migration and relocation choices of households; the workplace choices of individual people; location and relocation choices of businesses as well as the real estate market. When paired with microsimulation models of activity-based transport the choices and behaviours of agents within the urban system can be modelled at the individual level. These synthetic populations of agents are based upon high-resolution data, where available, or can be generated from more aggregate data using statistical techniques and stochastic variation.

The improved behavioural validity of microsimulation models comes at the cost of practicality. Data requirements both in terms of quantity and quality are greatly increased where inputs which are not available at the individual level must be generated increasing the time taken for model configuration. Execution times are also increased significantly which when combined with the stochastic variation across model outputs makes it a more complex task to generate and compare results for a range of scenarios.

3.4.4 Cellular Automata Models

In common with microsimulation models, Cellular Automata (CA) employs agents and actions at the microscale which when extended over time and space, produce aggregate behaviour. CA are mapped to a (generally two-dimensional) raster grid of cells and use local interactions to apply a provided set of rules to each cell in turn; this produces a changed grid which is swapped with the initial grid in the next iteration hence extending the application of rules over time and space (Clarke, 2014). The application of CA to model urban growth is widespread and assumes that future patterns of land use can be derived from localised interactions between land use of different types based upon past development. This can involve refinements from simple models to include the use of nonuniform cell spaces, irregular timesteps and complex transition rules (Sante et al, 2010).

The bottom-up nature of CA models, transmitting change via localised neighbourhoods of cells can lead to problems configuring such models using top-down policy drivers. This becomes particularly important when trying to capture important factors from higher spatial scales such as accessibility to employment. CA model execution including calibration is an inherently slow process which limits applicability in some cases.

3.4.5 Summary

Micro-simulation approaches are not well-suited to the task of rapidly assessing future scenarios of land use and transport in spite of their improved behavioural validity and greater spatial disaggregation. This is due to significantly increased demand for input data, issues with stochastic variation and very long model execution times.

The comparative reduction in complexity and model execution times afforded by aggregate spatial interaction and aggregate utility models provide potential solutions to the requirement of rapid assessment. When comparing these two approaches it must be questioned whether the improved representation of locational choice afforded by utility-based models warrants the increase in model complexity and data requirements. It can be argued in the case of climate change mitigation and adaptation that models of land use and transport should rely less on the consideration of preferences but instead focus on fundamental needs (Wegener, 2014).

A key problem with aggregate approaches to modelling land use and transport is the spatial representation of such systems. Trips between zones assume that activities are located at zonal centroids, and the land use resulting from such models is not spatially explicit beyond an aggregate value for each zone. This “Tyranny of Zones” (Spiekermann and Wegener, 1999) means that land use attributed to population increase cannot be meaningfully compared with environmental models such as air quality which are represented at fine spatial scales, typically as raster grids.

One way of solving the problem of aggregation is to adopt a multi-level and multi-scale approach as shown in figure 3.5 for the Dortmund region where multi-level refers to economic downscaling from global projections to the region of interest, and multi-scale refers to the consideration of the urban area as both a collection of administrative zones with inter-zonal processes, and as an arrangement of regularly-spaced grid cells with intra-zonal proximity attractors. The advantage of such an approach is that each process can be represented and modelled at its own spatial-scale which allows for the combination of top-down (aggregate) with bottom-up (disaggregate) processes which are hierarchically constrained. Care must be taken to ensure consistency across modelling levels and spatial scales, for instance, the representation of inter-zonal accessibility to employment via transport networks in the zonal system should be mirrored by the proximity to transportation in the raster grid system.

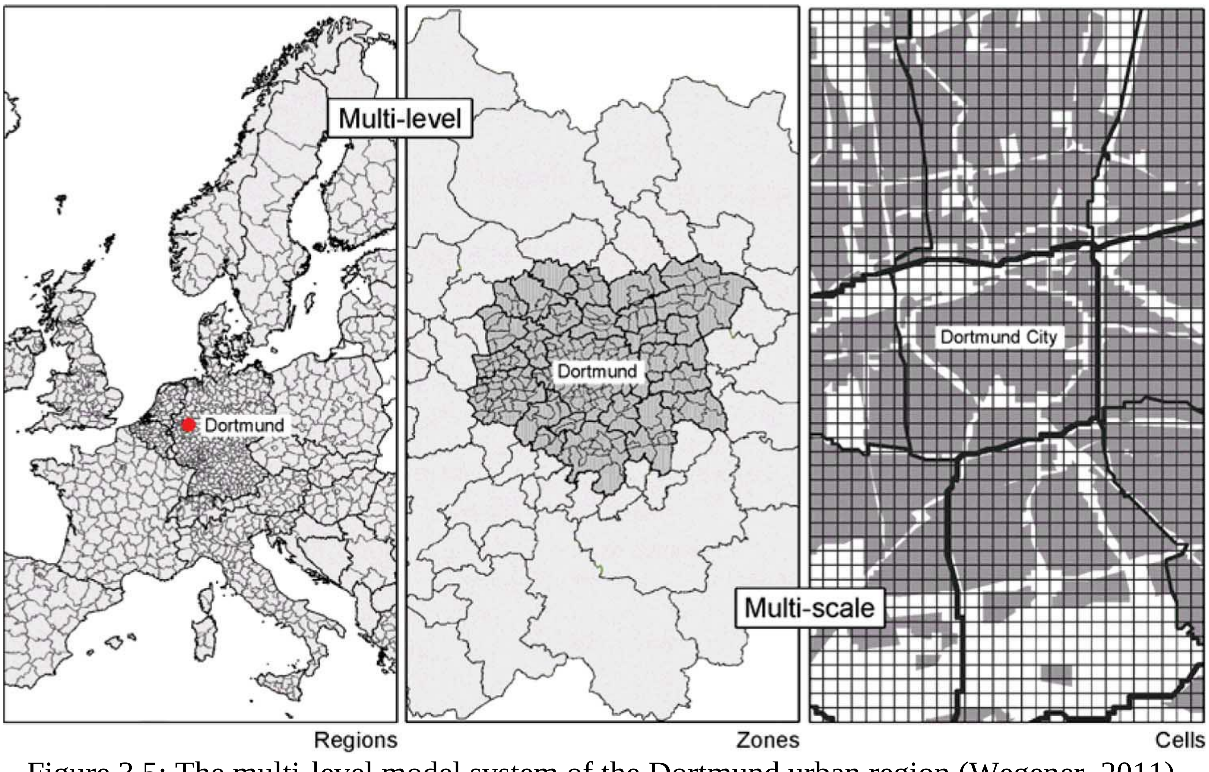

Figure 3.5: The multi-level model system of the Dortmund urban region (Wegener, 2011).

3.5 Urban Integrated Assessment Framework

The aim of this research is to develop a simple, transferable and consistent modelling system suited to the purpose of generating plausible scenarios of future development for analysis along with environmental factors. Building such a system from first principles in a relatively short timeframe is not feasible so an existing framework was identified which has desirable properties but does not meet all of the requirements of integrated modelling frameworks for stakeholder dialogue as described in section 3.3

The Tyndall Centre's Urban Integrated Assessment Framework (UIAF), (Hall *et al*., 2009) integrates land use, transport, and environmental models across consistent scenarios to consider the mitigation and adaptation of the urban system across multiple scales. This facilitates the city-scale assessment of potential future interaction with climate hazards such as extreme heat, water scarcity and flooding. The UIAF models changes in population and land use in scenarios consistent with emissions and climate impacts enabling the concurrent testing of mitigation and adaptation strategies to examine and compare consonance and dissonance between policies.

Changes in land use and transport driven by socio-economic processes are linked to long-term climate impacts for the greater London area in scenarios driven by exogenous projections of employment, population and climate hazards. An econometric model, E3MG (Global Energy-Environment-Economy Model) provides drivers based on the global economy using a variety of indicators including oil prices to provide future growth projections. This is downscaled at a national level to attribute growth to a particular region using a multi-sectoral dynamic model of the UK (MDM).

The UIMF takes these higher-scale models as exogenous drivers for the urban area and further downscales employment, population and land use in two stages from urban area through urban zones to urban land parcels composed of raster grid cells. An important consideration for model use is the application of data produced at each spatial scale; growth projections for the entire urban area based upon economic and population growth can be used to produce emissions data based upon aggregated data, administrative decisions such as the provision of school places can be based upon zonal data, whilst interactions with environmental models can only be explored by further spatial disaggregation to the urban-parcel scale.

The methodology developed by this research unifies the multi-scale LUTM components of the UIAF in an integrated modelling framework. The UIAF was developed in a modular fashion by separate teams using their own preferred software and languages, these modules are integrated in a process described at best as loose coupling to simulate transport, locational choice and land use in the Greater London area.

The requirements of integrated modelling frameworks for stakeholder dialogue as described in section 3.3 are: **Validity**, providing confidence in model outputs; **Transparency**, documenting model processes, equations and algorithms; **Usability**, simplifying model calibration and scenario generation; **Flexibility**, allowing models to be grouped and their execution controlled in a range of configurations; and **Performance**, ensuring fitness for interactive stakeholder dialogue. These requirements are used to critique the UIAF in the following subsections to define design criteria for an integrated modelling framework presented in the next section.

3.5.1 Validity

A spatial interaction model (SIM) uses accessibility to employment via transport networks to simulate population scenarios resulting from land use and transport planning policy. A set of weighted zonal attractors is incorporated to represent a range of spatial factors derived from planning policy decisions whilst a transport accessibility model (TAM) uses generalised cost to evaluate trips allowing for network modification to test scenarios of transport investment and infrastructure development. The urban development model (UDM) maps zonal population to a raster grid using attractors and constraints representing planning policy decisions. This increased spatial detail of development in the UDM permits the assessment of exposure to spatial climate hazards. The UDM is based upon multi-criteria evaluation (MCE) of development suitability and a process similar to cellular automata (CA) to spread development across and within land parcels.

The outputs of this modelling arrangement are multi-resolution rather than multi-scale given that both the SIM and UDM are applied to the same region. Although both models are parameterised with global data for the study region there is a clear distinction as to the spatial extent of each modelling stage. The SIM models the inter-zonal disaggregation of people and jobs based upon employment accessibility via the transport network, whereas the UDM models the resulting land use from locational choice based upon localised factors such as proximity to transport. A simple view of the modelling arrangement is of a nested set of

UDMs, one for each zone in which each UDM is spatially bound by zonal extents whereas SIM operates across the zonal system and is bounded by regional extents.

The coupling of SIM and UDM is a key process in terms of capturing micro-scale dynamics within an aggregate model of locational choice; inter-zonal attractors in SIM are replicated by intra-zonal attractors in UDM for example, the transport accessibility in SIM is replicated by proximity to transport in UDM. This results in a consistent modelling system which considers two spatial scales in a simplified manner resulting in fine scale development outputs from minimal input data.

3.5.2 Transparency

The relatively simple models used in the UIAF can be conceptually described in terms which are easily communicable, making them suitable for interactive engagement with stakeholders and the public. The model implementations themselves along with their integration within the framework is somewhat more complex but must be adequately described to potential users. Developers may wish to modify any part of the framework or individual models requiring full code transparency meaning that open source software only should be used.

3.5.3 Usability

The UIAF has no interface and other than by rewriting model code there is no way of modifying its operation or configuring scenarios. The minimum set of data required by the UIAF is relatively modest but models can make use of more input data, where available, such as attractors in the SIM and UDM to represent additional spatial drivers of population and land use change. To maximise the potential user-base of the framework, methods using open source software to prepare model inputs should be documented which should be supported by data handling functions to map inputs and parameters to framework models.

3.5.4 Flexibility

Different scenarios require different drivers in terms of model inputs; in the UIAF these alternatives are provided as model-specific configuration files in which outputs from the previous modelling stage are stored as alternative inputs to the current model. This approach requires that each model is responsible for selecting an input from a specified range of provided datasets meaning that modelling stages must be configured in cooperation to generate the desired result. This collective configuration across models implemented in different languages places an unnecessary burden on the user who simply wishes to consistently drive a modelling scenario. The framework should simplify the task of consistent configuration across models by formalising model interfaces and abstracting the process of scenario selection. Exploratory modelling requires that multiple scenarios are run to assess the implications of alternative policies. Uncertainties and sensitivities can be explored by executing the model multiple times whilst varying inputs. To facilitate this, the framework should allow models to be grouped and executed in a range of configurations, providing control across iterations to adjust inputs, outputs and parameters.

3.5.5 Performance

The emphasis on rapid assessment suitable for stakeholder dialogue places a limit on model execution times. Although it is difficult to ascertain the overall performance of the UIAF comprising multiple languages and modes of execution, a significant bottleneck can be found in the computation of shortest paths between origin and destination nodes in transport networks using Dijkstra's algorithm. The prototype code for shortest-path computation uses $networkx$, a purely python implementation which stores lists as python dictionaries which are inefficient in-memory containers. The algorithm performs poorly for large networks and renders the model unsuitable for interactive purposes. In more general terms, the required flexible model runtime must be combined with higher performance model computation.

3.6 Urban Integrated Modelling Framework

The principal aim of this research is to package the LUTM components of the UIAF into a transferable and consistent modelling system suited to the purpose of generating plausible scenarios of future development for analysis along with environmental factors. In the first step of this process, the LUTM scenario generation system is separated from the environmental impact models resulting in the system diagram as shown in figure 3.6. This collection of models along with their implementation in the specific software environment developed by this research is henceforth referred to as the Urban Integrated Modelling Framework (UIMF) reflecting the separation of LUTM scenario generation from environmental impact assessment, and the focus on flexibility, performance, transferability and consistency.

The requirement for transferability dictates that the framework should be portable across study regions determined by different model parameters. The proof-of-concept integration of models created by development teams in the UIAF was achieved by loosely coupling models configured for consistent scenarios based on a shared study region, the Greater London area, resulting in a framework which is bound to its original application area and to the agreed scenarios. If the framework is to be applied to new study regions without the collaborative involvement of the original model development teams it must be decoupled from the Greater London area. Decoupling aims to extract key model parameters such as the number of zones or available modes of transport allowing the framework to be transferred between study areas and scenarios. This involves the removal of hardcoded variables, control flow parameterisation, and handling external model configuration data in a variety of file formats.

Figure 3.6 shows the modelling system to be implemented in the UIMF. From left to right, the operation and flow between models is as follows:

1. For each mode of transport, network data for is combined with generalised cost parameters considering both money and time. Least-cost paths are found for each mode before combining costs across all modes and converting via a deterrence function to an origin-destination matrix of transport accessibility.
2. In each timestep, projected values of employment and population for the urban area are spatially disaggregated into zones using zonal attractors and constraints where zonal population is also based upon employment locations and accessibility to that employment via the origin-destination matrix of transport accessibility.

3. Zonal population is further spatially disaggregated to a raster grid of cells using rasterised local attractors and constraints producing a detailed map of current and potential future land use.

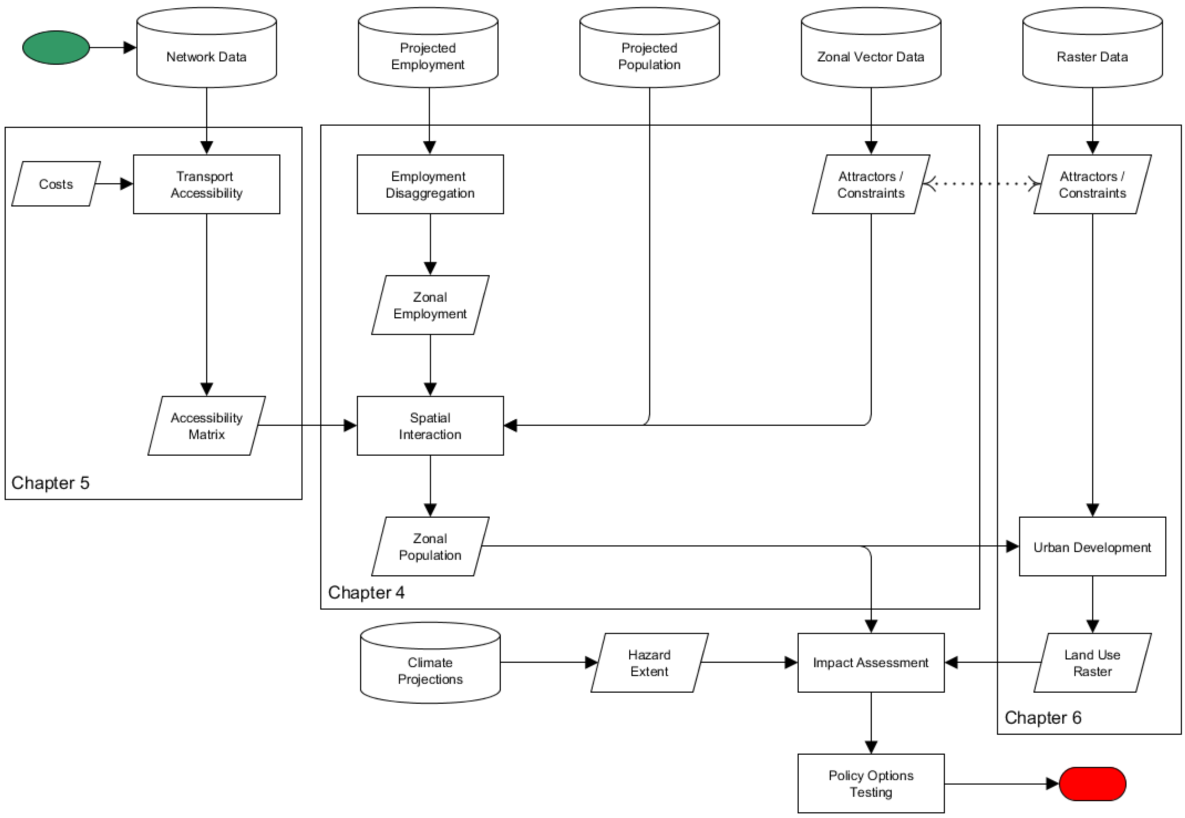

Figure 3.6: Models implemented in the Urban Integrated Modelling Framework

Each model in the system shown in figure 3.6 is reimplemented using generic features of the Urban Integrated Modelling Framework shown in figure 3.7. These implementations are described in detail in the following chapters: Spatial Interaction Model (SIM), chapter 4; Transport Accessibility Model (TAM), chapter 5; and Urban Development Model (UDM), chapter 6.

Figure 3.7 gives an overview of the software architecture of the UIMF which was designed in response to the critique of the UIAF in sections 3.5.1 to 3.5.5 in terms of requirements for modelling with stakeholders described in section 3.3. Features of this design indicated in figure 3.7 are discussed in the following subsections:

- 3.6.1 describes the use of a database to store and manage spatial datatypes corresponding to the model inputs shown across the top of figure 3.6.
- 3.6.2 describes the use of a minimal standardised model interface using tables of inputs, outputs and parameters.

- 3.6.3 describes simple model coupling as part of a model group.
- 3.6.4 describes advanced flow control across model groups in support of situations requiring multiple model runs.
- 3.6.5 describes the flow of model data and metadata.
- 3.6.6 describes model execution and computational flexibility.

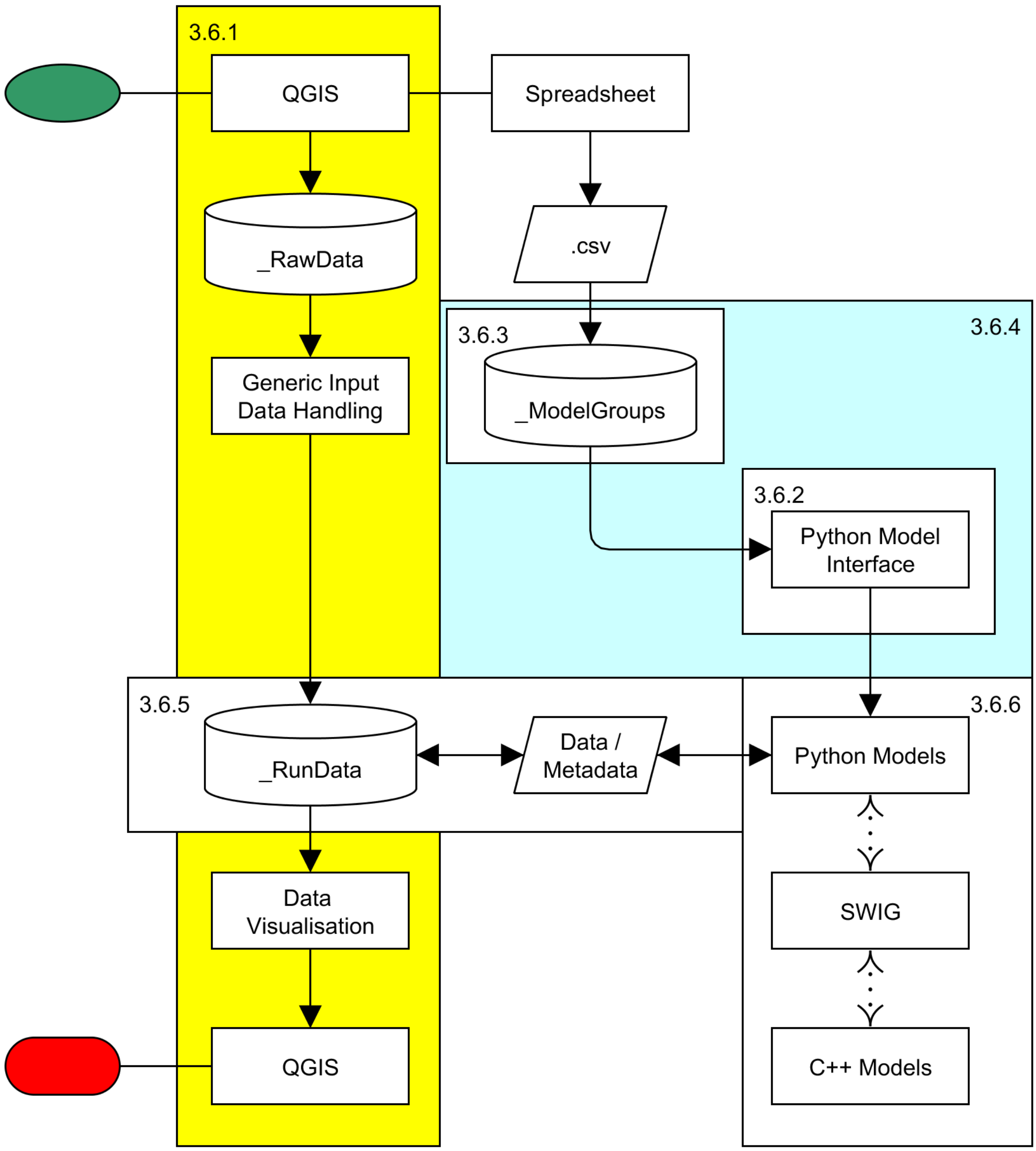

Figure 3.7: Software Architecture of the Urban Integrated Modelling Framework

3.6.1 Spatial Data

The suite of models shown in figure 3.6 describe the geography and interactions of an urban system using the following spatial datatypes:

- Polygons
    - The study region is described by a collection of polygons which defines the extent of each zone in the SIM. Polygons also define parcels of land with particular attributes which can be used as attractors or constraints such as currently developed land.
- Points
    - Accessibility in the TAM models journeys between origin and destination zones using point locations (network nodes) at the centroid of zone polygons. Points are also used to generate proximity rasters in the UDM for features of interest such as public transport stations.
- Linestrings
    - Routes in the transport networks used in the TAM are represented by linestrings (network edges).
- Cells
    - The UDM describes the study region as a grid of uniform cells, each of which is typically 1 hectare in area. Vector features such as polygons representing zones, attractors and constraints in the SIM must be rasterised to ensure a consistent spatial representation across models.

PostgreSQL is an established, free and open source object-relational database management system which supports Structured Query Language (SQL) and can be extended to support geographic objects and spatial queries using PostGIS. The resultant spatial database can be managed using pgAdmin which is a web or desktop application allowing database exploration and advanced administration. QGIS is a free and open source Geographic Information System (GIS) which can connect to spatial databases and allows geospatial data to be edited, analysed and visualised. The UIMF uses all of these tools to manage spatial data from the preparation of model inputs to the visualisation of model outputs as shown in the left column of figure 3.7.

Input vector data can be prepared using QGIS or uploaded to the spatial database via shapefiles in $pgAdmin$ using the shapefile import/export manager plugin. The use of ascii raster files provides a simple format to exchange raster data along with the following header information:

- $ncols$ (number of columns)
- $nrows$ (number of rows)
- $xllcorner$ (lower-left-corner, lower-left-cell, x coordinate)
- $yllcorner$ (lower-left-corner, lower-left-cell, y coordinate)
- $cellsize$ (size of each cell)
- $NODATA\_value$ (cell data mask value)

The ascii format is used as the UIMF raster interface in terms of importing and exporting data which can be prepared and viewed in QGIS but since a large amount of raster data may need to be transferred between models and to/from the spatial database some modifications have been made to improve performance which are described in Appendix A.

The need for the UIMF to be portable between study regions in different parts of the world requires that a Spatial Reference System (SRS) identifier is used to define the map projection and hence the location of geographical entities. The value used for British National Grid is 27700 for example.

3.6.2 Python Model Interface

As well as handling spatial data as described in the previous subsection, the UIMF uses PostgreSQL to manage tables of non-spatial data defining each model or sub-model to be executed by the framework in minimalist terms. This standardised model definition is in contrast to the use of bespoke model configuration files and operations as was used in the UIAF model implementations.

All models within the UIMF are run via the Python Model Interface (PMI) shown in figure 3.8 which requires that each model is described by a set of database tables holding inputs, outputs, and parameters which can be created and modified using any comma-separated values (CSV) file editor. The Python script which executes each framework model can then establish a connection to the database holding these tables, use parameters to configure model operation, read inputs from named database tables, run the model code and write outputs to named database tables.

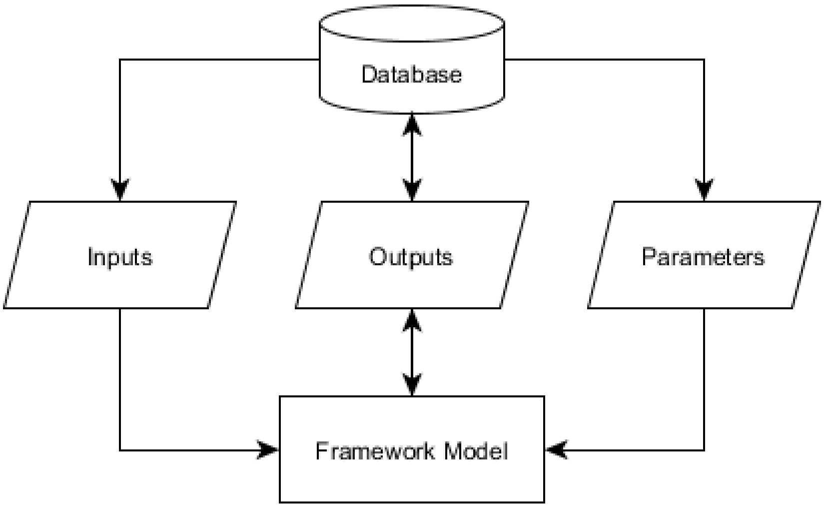

Figure 3.8: Python Model Interface

In all models running under the PMI, a Python module which executes the model code is linked to database tables specifying model inputs, outputs and parameters. Tables of inputs and outputs share the same format (table 3.4), consisting of an integer primary key along with string columns for names and values. Parameter tables (table 3.5) consist of an integer primary key, an integer model key and named columns of user-defined type for *n* parameters as required.

primary key	**name**	**value**
1	data_1	table_1

Table 3.4: Example PMI model inputs / outputs

primary key	**model key**	**parameter_1**	**parameter_2**
1	1	1	234.5

Table 3.5: Example PMI model parameters

The model to be run is linked with its datasets via a database table specifying the model group (table 3.6) which consists of an integer model key and a string column for the model and each of its generic datasets. The model column entry names the Python module to be run whilst the columns for inputs, outputs and parameters reference configuration tables stored in the same database.

model key	**model**	**inputs**	**outputs**	**parameters**
1	m1	m1_inputs	m1_outputs	m1_parameters

Table 3.6: Example PMI model group

The PMI model class has the following key methods:

- $load()$ - load the model into memory along with all inputs, outputs and parameters
- $run()$ - execute the model

The PMI can interact with and run a model if it is executed via a main function in the Python module which takes Python dictionaries for inputs, outputs and parameters as arguments; this function should unpack the provided dictionaries of pointers, manage the transfer of data and run the model before returning a Boolean value which is used to control model iteration.

3.6.3 Model Coupling

All models in the UIMF are executed via a model group as shown in table 3.6 in which each model has its own row. Models can be coupled together in a simple manner by specifying the order of model execution and routing specified outputs to inputs as illustrated in figure 3.9.

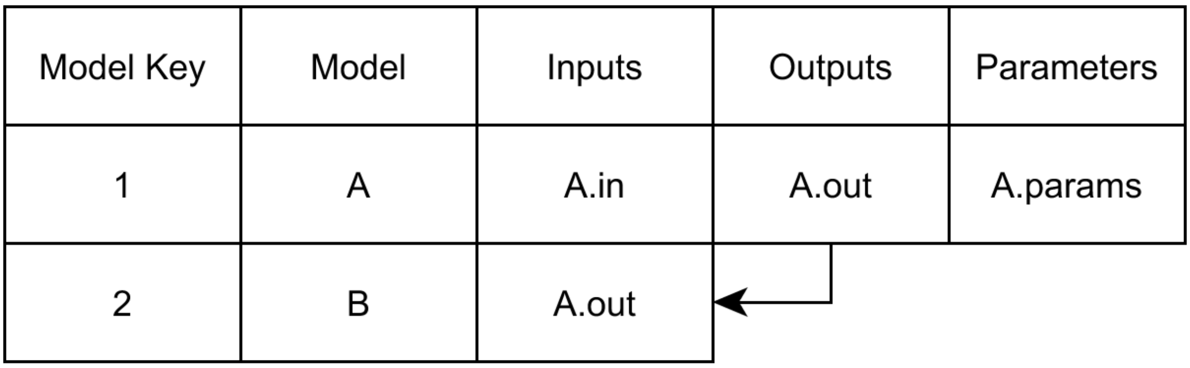

Figure 3.9: Simple Model Coupling Example.

The PMI model group class has the following key methods:

- $readtable()$ – creates a model list from a database table
- $rungroup()$ - loads and runs each model in a model list
- $iterate()$ - calls $rungroup()$ according to $n\_iterations$

Within a group, models are run sequentially via a Python script whose main function is executed when Python calls the module from the command line; this function uses a Python-PostgreSQL database adapter to form a connection to the database specified by $connect()$ function arguments before interfacing with the model group class. The model group

$readtable()$ function is called with the database connection and specified model group table as arguments before setting the $n\_iterations$ value and calling the $iterate()$ function.

The PMI provides three modes of iteration, all of which run each model in the group. The desired mode is specified using the $n\_iterations$ value as follows:

- Group (0) run group once
- N-conditional (n) run group n times unless stop condition is reached
- Forced-conditional (-1) run group indefinitely until stop condition is reached

In the first mode the execution is dependent solely upon the number of rows representing models, sub-models etc. in the model group list. The remaining modes use the Boolean value returned by the main function of each model executable to terminate iteration of the group if any model returns a value of True indicating that the stop condition has been reached.

The flow of operations in the PMI as depicted in figure 3.10 is as follows:

- The $readtable()$ function creates and populates a model list from a PostgreSQL database provided by the user
- The $iterate()$ function iterates over the model list according to the $n\_iterations$ variable, calling the $rungroup()$ function in each iteration mode. If the $group.stop$ variable is True, then iteration of the model list stops
- The $rungroup()$ function runs each model in the list using the model class $load()$ function to load the model and its datasets into memory, and the $run()$ function to call the PMI compliant main function in the Python module. If the Boolean $model.stop$ value returned by the main function of the model is True, then the $group.stop$ variable is set to True

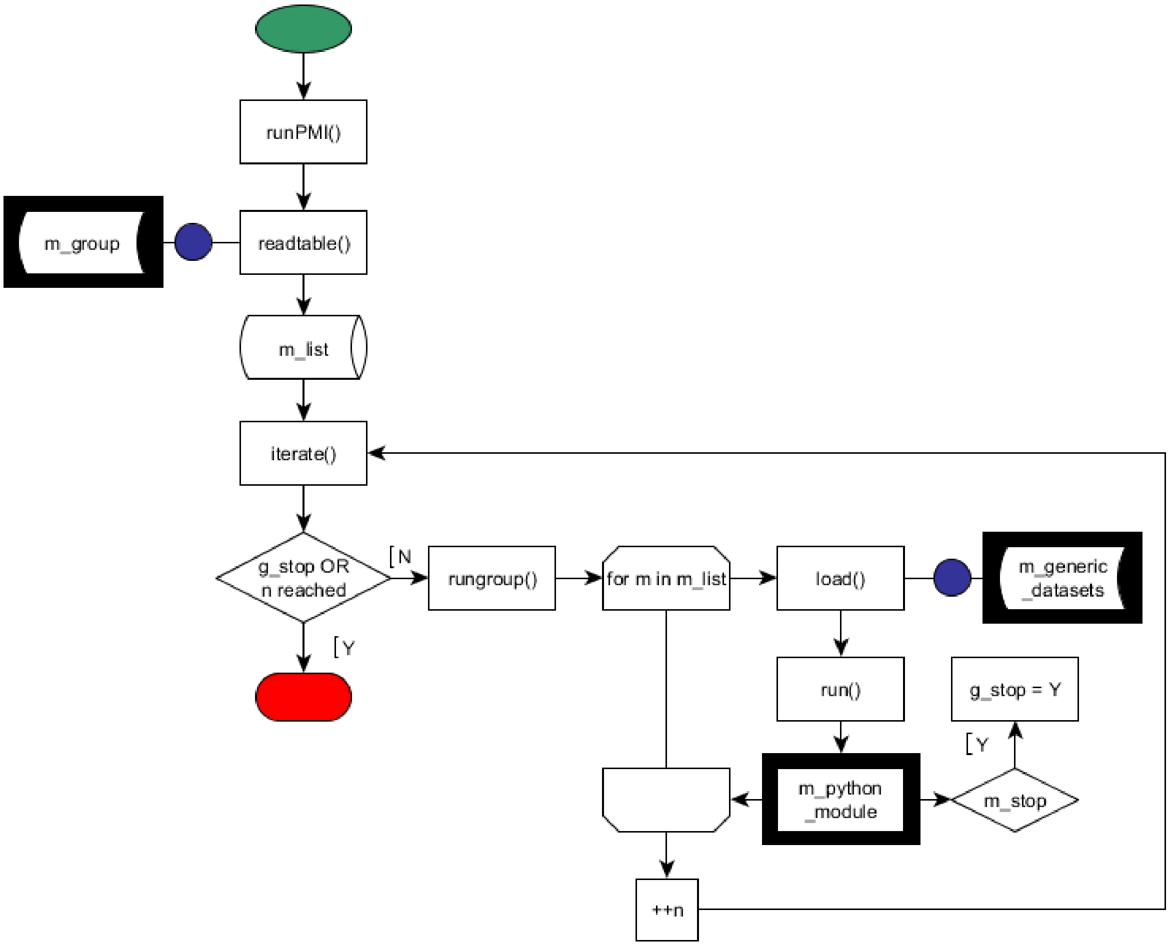

Figure 3.10: Running a model group via the PMI using the n-conditional iteration mode; user-provided database tables and model shown in black.

3.6.4 Advanced Flow Control

The previous section described how the PMI can be used to iterate over a group of models in a flexible manner. Using the simplest form of iteration, a model can be run multiple times within the same group by specifying it in multiple rows of the model group table (Table 3.6), where each instance of the model has a different model key. The model parameters table can then be used to provide a different set of values for each model instance as specified by the model key column. This arrangement can be used to generate a range of model outputs, for example, based upon swept parameters for uncertainty analysis.

The different parameter values used in each instance of the model generate different model outputs which should be stored in their own database table. This mapping of model iterations to output tables could be achieved by deriving the name of output tables using a suffix indicating the model instance; however, if the same model code is to be reused in different applications then this mapping should be external.

The UIMF uses model drivers to generalise the control and mapping of model instances within model groups. A model driver must be able to be run as part of the model group and therefore must adhere to the requirements of the PMI. The key feature of a model driver is that it can access and modify the datasets of other models within its group thereby managing the routing of inputs and outputs, and the use of parameters both within the group and across group iterations. This allows the models themselves to be fixed in their functionality which in turn facilitates the ability to arrange and rearrange models without modification of their module code.

The generic methods presented so far show how models can be chained in model lists controlled across iterations by model drivers. The rationale for grouping models in any given way varies according to the requirements of the simulation run; models may be configured such that the group is run until some condition is reached (e.g. equilibrium), or to produce results for a swept range of input values.

The UIMF accommodates scenarios involving multiple model groups, with each group performing a given task within the overall simulation. This is achieved by allowing the user to specify a group of model groups to be executed. This group of groups is stored in a database table (table 3.7) containing a string column to list model groups and an integer column to specify the corresponding iteration control for each group specifying the value of $n\_iterations$ to be used.

model group	**group iterations**
g1	0

Table 3.7: Group of groups table format

The python script to run the framework can then be configured to export the named group of groups table to a CSV file before reimporting and performing the following for each row in the table:

- Use the provided entry in the model group column as an argument to the $readtable()$ function
- Set $n\_iterations$ to the value provided in the group iterations column
- Call the model group $iterate()$ function

The generic methods described in this section extend the PMI to allow for the specification of a range of modelling scenarios. Using these foundations, the user can build model group chains where each model group has its own iteration control and model drivers to control data routing.

3.6.5 Data and Metadata

When models are chained to form an overall system simulation the interface between models i.e. model inputs and outputs must be recorded so that variation in results can be attributed to variation in model inputs; this requires a metadata system which is chained accordingly to keep track of model configuration. The framework should support future model development by using a minimally prescriptive model format along with generic model datasets; this should simplify coupling newly developed models with existing models whose key parameters are extracted. The exploration of model uncertainties based on the automated generation of large result sets with accompanying metadata should also be supported.

The results produced by any given model run can only be of lasting value if the means of their production in terms of the model and its data are preserved. Model data should be retained and made available to users of the resulting models and data. The PMI requires that the main function in a compliant model executable should unpack the provided dictionaries of data pointers to use that data. To record metadata, the main function should also repack the provided dictionaries i.e. the inputs, outputs and parameters, and store this in a database table for use in conjunction with the model output for analysis.

To simplify the table format, all metadata is stored as a pair of string variables with columns for names and values. The process of recording metadata within a Python module begins with the creation of two Python lists: ‘name’ and ‘value’. These lists are then filled with corresponding data for each item described in PMI model datasets using either $append()$ to add single items, or $extend()$ to combine lists of items. Once this process is complete, the ‘name’ and ‘value’ lists are used to form a ’metadata’ list of lists which is combined in a single list of name-value tuples using the $zip()$ function.

When models and sub-models are arranged in a model chain where the output of one model serves as the input to another and so on, it is important that the final model metadata contains the metadata for all models before it in the chain to comprehensively describe the model run configuration. When forming metadata chains any model whose input is derived from another model output should assume responsibility for retrieving and incorporating metadata from the previous stage. The list of metadata for the current model or sub-model is added to the end of the input metadata list using the $extend()$ function and written to a newly created database table e.g. $m2\_output\_md$, as before.

A model which is preceded by a model driver should be indicated by a value of $true$ for the generic model parameter $driven$. The process of retrieving and incorporating the metadata from the driver model named in the generic model input $driver\_name$ can then be automated As with chained model metadata, the list of metadata for the current model or sub-model is complete it is added to the end of the driver metadata list using the $extend()$ function.

3.6.6 Model Execution

The UIMF makes use of SQL commands embedded within Python programs to interact with model data stored in spatial database tables. These Python modules can be arranged and connected in a flexible manner to meet the requirements of a range of modelling scenarios. Interactivity and flexibility are two of the key attributes of Python and scripting languages in general, another benefit of using Python is the ability to make use of existing libraries written in C/C++ where improved performance is required.

The Python Application Programming Interface (API) details the steps required to manually extend Python with modules written in C/C++; however, this is a laborious task which would need to be repeated for each new extension. The key principle of language interoperability in this case is to convert function arguments from Python to C/C++, and the results returned by functions back from C/C++ to Python. The Simplified Wrapper and Interface Generator (SWIG) was chosen to accomplish this task within the UIMF due to its ease of use.

SWIG is a software development tool which creates wrappers to interface C/C++ programs with several different languages including Python; it provides extensive customisation options but often requires only basic information to package C/C++ libraries for use as Python extension modules. Wrappers are generated using interface files in which everything to be included in the extension module is declared. These interface files can simply reference existing C/C++ headers for function prototypes and class definitions. The declarations in the interface file are used to generate the 'module' which has two outputs: a $\_wrap.cxx$ file which is compiled into a shared library, and a $.py$ file which is imported by users of the module. The final extension module (including the $\_wrap.cxx$ file) is compiled into a shared library using a file format ($.pyd$) which is the Python equivalent of a Dynamic Link Library (DLL).

The management and transfer of data between spatial database tables and models which consume and produce that data should be handled within the Python module responsible for executing each model. The UIMF provides numerous functions and techniques to manage this data handling process which involves vector, raster and non-spatial data types. In the case of using wrapped C++ models, this exchange involves exporting data to and importing data from CSV files in a designated folder.

3.7 Summary

This chapter identified the need for an integrated approach to modelling future land use, transport and the environment (Section 3.2), before defining desirable characteristics required by such a system in Section 3.3. It then reviewed existing urban modelling frameworks and considered their relative strengths and limitations (Section 3.4). Section 3.5 presented the concept and models of the Urban Integrated Assessment Framework (UIAF) before critiquing the framework in terms of requirements identified in Section 3.3. In Section 3.6 the conceptual design and implementation of an Urban Integrated Modelling Framework (UIMF) was presented which reworks and integrates existing models in order to address problems identified in Section 3.5. The following three chapters detail the implementation of models within the UIMF beginning with the SIM in chapter 4.

Chapter 4. Spatial Interaction Model

4.1 Introduction

This chapter presents the Spatial Interaction Model (SIM) implemented in the Urban Integrated Modelling Framework (UIMF). Section 4.2 provides theoretical underpinnings, compares the SIM with similar models and justifies the simplified approach taken. Section 4.3 describes the development of this model from the reference implementation in the Urban Integrated Assessment Framework (UIAF) and highlights key differences. Section 4.4 gives a high-level overview of the model and its operation within the software environment of the UIMF, and Section 4.5 provides a detailed description of the SIM implementation.

4.2 Overview

Spatial interaction models the flow of activities between geographical locations often depicted by point locations representing larger areas. Flows between locations can represent a range of factors including traffic, migration and materials; however, focus is placed upon interactions which are routine and repeated. The physical networks by which interactions take place are not always modelled, instead a function of the distance between locations is often used as a measure of separation. Spatial interaction models are widely used as part of the four-stage transport model, modelling transport demand from existing land use.

Hansen (1959) demonstrated that accessibility and land availability can be used to model residential land use. This co-dependent link between land use and transportation paved the way for coordinated planning. Lowry (1964) developed the first operational model integrating land use and transport using nested spatial interaction models to simulate residential and service/retail employment locations within a zonal system.

The simplest form of spatial interaction models trips, $T_{(i)(j)}$, between residential locations and employment locations according to attributes of the origin, $V_{(i)}$, attributes of the destination, $W_{(j)}$, and their geographical separation, $S_{(i)(j)}$; this is formulated as in Eq. 4.1.

$$T_{(i)(j)} = f\left(V_{(i)}, W_{(j)}, S_{(i)(j)}\right) \text{ Eq. 4.1}$$

The SIM is singly constrained meaning that employment locations are established prior to the distribution of population. The potential for spatial interactions in any given zone to access employment via transport networks is a key driver of residential location choice. Employment and population models are coupled by sharing area availability within each zone, so each iteration of the employment/population model modifies available land for the next.

4.3 Development

The SIM is based upon the corresponding Zonal Model (ZM) from the UIAF which was implemented using MATLAB for the Greater London Authority (GLA) region, this was loosely coupled with other models in a climate impact assessment framework. A key requirement of the UIMF is that models should be transferable from one study region or scenario to another without modification of the source code; this means that the approach taken in the UIAF which was to create bespoke scripts and data-loading for each model run is not suitable. The SIM is generalised in terms of inputs, outputs and parameters and all model data is held in a PostgreSQL database.

All hardcoded parameters were moved from model scripts to database tables allowing the model to be applied to case studies with different numbers of zones, timesteps, population attractors, employment attractors and sectors of employment; this enables the SIM to allocate memory and load input data for different model applications. All model inputs and outputs were moved from hardcoded local files to named PostgreSQL database tables enabling model parameterisation via a shared data reservoir.

The ZM uses exogeneous models to provide employment and population inputs for the GLA region. SIM generalises the process of model driving via projected values of employment and population. Clearly it is desirable to drive the SIM with projected values for both employment and population for the study region based upon reasonable assumptions, but it is also necessary to produce model outputs in cases where input data may be limited.

SIM can be driven using projected values of employment, population or ideally both. In the case that either employment or population data is not available the SIM uses a fixed inverse activity ratio calculated using observed values of employment and population to estimate the missing set of projected data.

Projected data for employment and/or population may not be available at the level of the study region therefore it is useful to be able to drive the model using either national data or simple growth functions. The SIM treats projections of employment and/or population as percentage drivers relative to observed values.

The ZM deals only with positive growth, in cases of decline the model either provides no response in terms of population or offsets declining sectors against growing sectors in terms of employment. This approach is incomplete and in the case of employment results in sectors being located in areas which do not necessarily correspond to their input attractors as a result of the offsetting process across sectors. The SIM explicitly models both positive and negative change in employment and population using reciprocal drivers to handle decline, this assumes that since positive change is spread in proportion to relative zonal attractiveness, negative change should be spread in proportion to relative zonal unattractiveness i.e. decline is apportioned to the least promising areas for growth. Where either employment or population is in decline, land is given back to the zone in proportion to the observed density of development, hence assuming that buildings are repurposed or demolished to make way for new development.

The process of enforcing area capacity constraints in the ZM is retained in this model, but the results of the process are now output from the model to be used as practical checks on whether all employment and population could be accommodated in each timestep. Since the ZM does not adjust density values to ensure this is the case, the model should be iterated with increasing values of density in cases where current density values cannot serve demand for employment and population; the results of the area capacity constraints test provide a measure to test against in such situations.

The process of generating the available area for development within each zone is changed in the SIM. The ZM calculates this on the basis of vector data for each zone whereas the SIM uses a value output from the Urban Development Model (UDM) (Chapter 6) to ensure consistency across spatial scales. In the SIM, the available area for development value is calculated according to the number of available cells defined by the constraint raster input to the UDM. This results in values derived from the number of cells (i.e. where UDM uses 100m grid cells the available area value is in units of hectares).

Finally, the SIM records metadata for its own inputs, outputs and parameters and for all models which precede it in the modelling chain; this metadata is passed using the name of the model output appended with '_md' and passed on to UDM along with values of population and employment for further modelling. A comparison between the UIAF ZM and UIMF SIM is shown in table 4.1.

Functionality	**UIAF ZM**	**UIMF SIM**
Programming Language	MATLAB	Python/C++
Data loading	Bespoke	Shared database
Parameterisation	No	Yes
Projected data adjustment	No	Yes

Decline	No	Yes
Overflow	No	Yes
Consistent with UDM	No	Yes

Table 4.1: UIAF ZM Vs UIMF SIM.

4.4 Framework Spatial Interaction Model

4.4.1 Overview

As shown in Figure 4.1 the SIM implemented in the UIMF is linked to PostgreSQL database tables via the Python Model Interface (PMI) described in Section 3.6.2 The SIM algorithm is written in C++ which is made callable from Python using the Simplified Wrapper and Interface Generator (SWIG). High-level operation of the model involves exporting data in the Python script from named input database tables to csv files held in the designated swap folder to provide inputs to the C++ model.

The parameters emp_driver and pop_driver determine whether projected employment and population datasets are available and therefore should be exported and used as model drivers. The C++ SIM class constructor is initiated by passing in key parameters such as the number of zones which allows the C++ model to allocate memory to load input data. Data is then loaded into the C++ model from csv files in the swap folder and the model is executed.

Upon completion of the C++ SIM algorithm, the Python script creates PostgreSQL database tables named as outputs via the PMI to store results which are copied from csv files held in a designated swap folder and model inputs, outputs and parameters specified by the PMI are gathered into SIM metadata. Since the SIM takes the output of the Transport Accessibility Model (TAM) as an input, TAM metadata is loaded and incorporated into the model metadata chain before writing to the named PostgreSQL database along with model results.

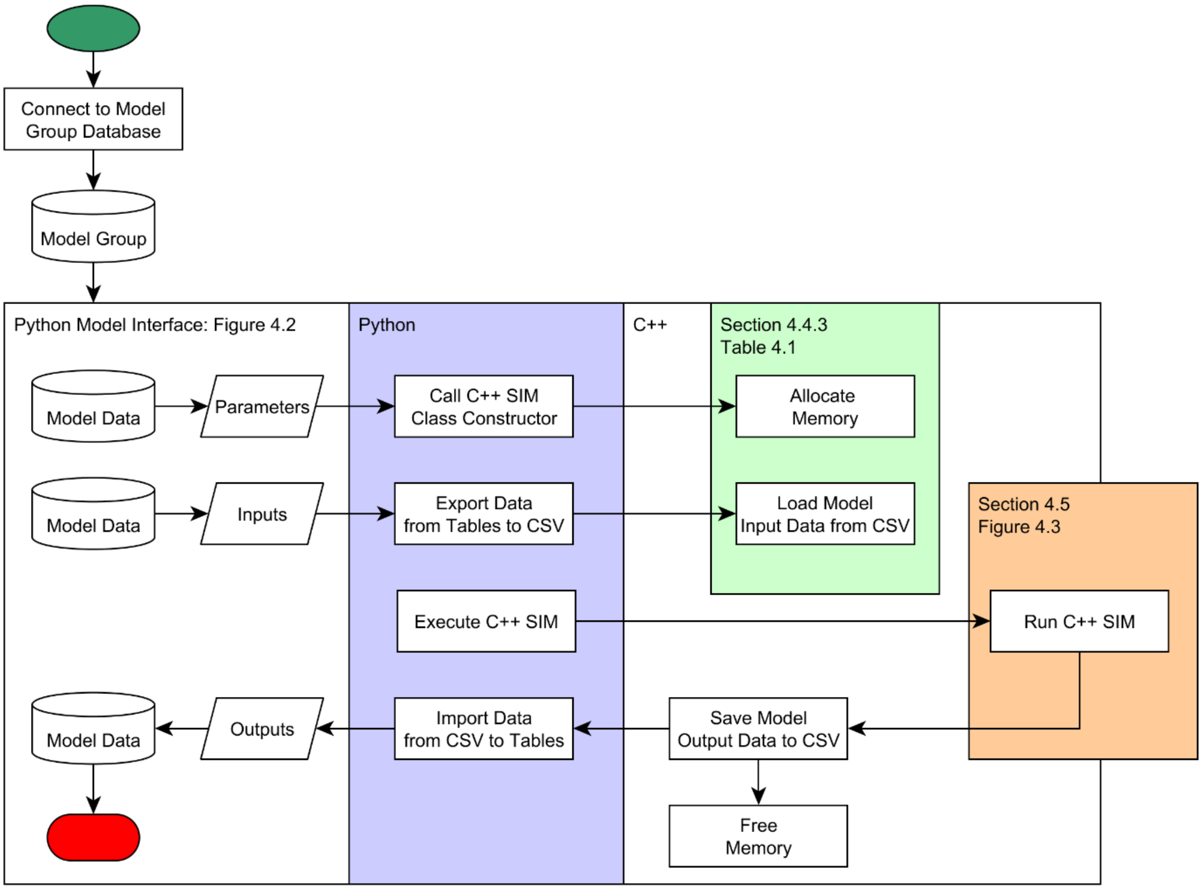

Figure 4.1: Overview of the Spatial Interaction Model framework.

4.4.2 Model Interface

An overview of the datasets for the UIMF SIM is shown in figure 4.2.

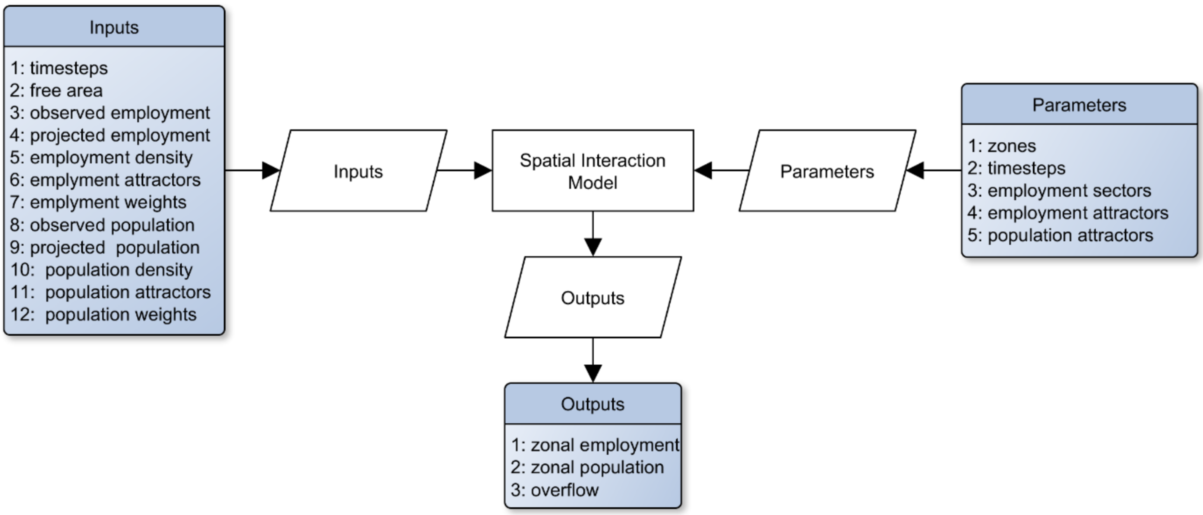

Figure 4.2: Spatial Interaction Model Data

Integer parameters shown in figure 4.2 specific to the zonal model are: **'number of timesteps' (t)**, **'number of zones' (z)**, **'number of sectors of employment' (s)**, **'number of employment attractors' (e)** and **'number of population attractors' (p)**. This parameterisation specifies the dimensions of the zonal model, its inputs and outputs; and is fundamental to transferability between study regions.

The inputs **'observed employment'**, $\boldsymbol{E^{obsrv}_{(z)(s)}}$, and **'observed population'**, $\boldsymbol{P^{obsrv}_{(z)}}$, describe the initial spatial distribution of employment and population i.e. in the first timestep. The input **'free area'**, $\boldsymbol{A^{free}_{(z)}}$, quantifies the land available for future development within each zone and can be described as in equation 4.2.

$$A^{free}_{(z)} = A^{total}_{(z)} - A^{constr}_{(z)} - A^{obsrv\ emp}_{(z)} - A^{obsrv\ pop}_{(z)} \quad \text{(Eq. 4.2)}$$

Where $A^{total}_{(z)}$ is the total area of the zone, $A^{constr}_{(z)}$ is the area subject to development constraints, $A^{obsrv\ emp}_{(z)}$ is the area which currently houses employment, and $A^{obsrv\ pop}_{(z)}$ is the area which currently houses population.

Modifying the constrained area, $A^{constr}_{(z)}$, and recalculating $A^{free}_{(z)}$, allows for a range of constraints to be applied reflecting different land use development policies.

The inputs **'projected employment'**, $\boldsymbol{E^{proj}_{(s)(t)}}$, and **'projected population'**, $\boldsymbol{P^{proj}_{(t)}}$, provide possible future values based on exogenous model inputs.

These values are converted to ratios which multiply $E_{(z)(s)}^{obsrv}$ and $P_{(z)}^{obsrv}$ respectively; this allows the change in employment and population within the study region to be driven in a flexible manner e.g. using national projections or percentage multipliers.

Future development i.e. where $t > 0$, is carried out according to the inputs **'employment density', $\boldsymbol{D_{(z)}^{emp}}$,** and **'population density', $\boldsymbol{D_{(z)}^{pop}}$**. Observed density values are calculated using equations 4.3 and 4.4 respectively, to which offsets and multipliers can be applied to reflect densification scenarios etc.

$$D_{(z)}^{emp} \;=\; E_{(z)(s)}^{obsrv} \big/ A_{(z)}^{obsrv\ emp} \quad \text{(Eq. 4.3.)}$$

$$D_{(z)}^{pop} \;=\; P_{(z)}^{obsrv} \big/ A_{(z)}^{obsrv\ pop} \quad \text{(Eq. 4.4)}$$

The spatial disaggregation of employment is driven by the inputs **'employment attractors', $\boldsymbol{Att_{(z)(e)}^{emp}}$,** and **'employment weights', $\boldsymbol{W_{(e)(s)}^{emp}}$**; for example, $E_{(z)(s)}^{obsrv}$ could be used to attract future employment in proportion to existing employment. The spatial disaggregation of population is driven by the inputs **'population attractors', $\boldsymbol{Att_{(z)(p)}^{pop}}$, 'population weights', $\boldsymbol{W_{(p)}^{pop}}$,** and **'accessibility matrix', $\boldsymbol{Acc_{(z)(z)}}$**; where $Acc_{(z)(z)}$represents the accessibility between origin and destination zones as output by the travel model. Both sets of attractors and weights are normalised during model execution to simplify the configuration process.

4.4.3 Setup and memory allocation

After an instance of the zonal model class is created, the setup function is called with the parameters **s, t, z, e** and **p**; these parameters are copied to integer variables used for memory allocation, data loading and iteration control. The memory footprint of the model is largely determined by the parameters which are not known at compile time, so dynamic memory must be allocated on the heap. The vector class within the C++ standard namespace could be used for this purpose; however, plain arrays are used in this case. All arrays are allocated in advance from the setup function and are deallocated when the model run is complete.

The method for loading input data (table 4.2) from .csv files employs arrays of string variables to temporarily store the contents of a single column of data read from file. Each parameter input to the setup function; **s, t, z, e** and **p**; has a corresponding dynamically allocated array; $S_{(s)}$, $S_{(t)}$, $S_{(z)}$, $S_{(e)}$ and $S_{(p)}$.

Name	Symbol	Rows	Columns
observed employment	$E_{(z)(s)}^{obsrv}$	z	s
observed population	$P_{(z)}^{obsrv}$	z	
free area	$A_{(z)}^{free}$	z	
projected employment	$E_{(s)(t)}^{proj}$	s	t
projected population	$P_{(t)}^{proj}$	t	
employment density	$D_{(z)}^{emp}$	z	
population density	$D_{(z)}^{pop}$	z	
employment attractors	$Att_{(z)(e)}^{emp}$	z	e
employment weights	$W_{(e)(s)}^{emp}$	e	s
population attractors	$Att_{(z)(p)}^{pop}$	z	p
population weights	$W_{(p)}^{pop}$	p	

Table 4.2: Input data dimensions and string arrays

Each array shown in Table 4.2 is loaded via a function taking the name of a .csv file as an argument. The data loading process makes use of two framework functions: $Extract()$ and $Convert()$. $Extract()$ reads a designated column of data from a named .csv file, storing it in a string array; $Convert()$ converts datatypes, from string to double in this case. To load the one-dimensional arrays of input data in table 4.2, $Extract()$ is called once to read data into the named **string array,** then $Convert()$ is iterated over **rows** to convert and copy each element of the array. This process is extended to load the two-dimensional arrays of input data by iterating over **columns**.

4.5 Detailed Spatial Interaction Model Description

Figure 4.3: Flow of Operations in the SIM.

4.5.1 Study Region Projections

The SIM is driven using projections of employment and/or population for *t* timesteps, in the initial timestep both employment and population are set to observed values. In all subsequent timesteps, projections are converted to ratios which are used to multiply observed employment or population. This ensures that discrepancies between observations and projected data in the initial timestep are resolved and allows the model to be used with a range of inputs including projections which are not specific to the study region and simple percentage drivers.

For each employment sector the employment ratio, $E_{(s)(t)}^{ratio}$, is found by dividing projected employment in each timestep by projected employment in the initial timestep (Eq. 4.5). Observed employment is then summed over all zones, z, for each sector, s, (Eq. 4.6) allowing the calculation of study region employment by multiplying observed employment by the employment sector ratio for each timestep and each sector (Eq. 4.7). The model of zonal employment (section 4.5.2) spreads the change in employment relative to the previous timestep so employment change for each sector in each timestep, $E_{(s)(t)}^{change}$, is found (Eq. 4.8).

$$E_{(s)(t)}^{ratio} = E_{(s)(t)}^{proj} / E_{(s)(t=0)}^{proj} \quad \text{(Eq. 4.5)}$$

$$E_{(s)}^{obsrv} = \sum E_{(z)(s)}^{obsrv} \quad \text{(Eq. 4.6)}$$

$$E_{(s)(t)} = E_{(s)(t)}^{ratio} \times E_{(s)}^{obsrv} \quad \text{(Eq. 4.7)}$$

$$E_{(s)(t)}^{change} = E_{(s)(t)} - E_{(s)(t-1)} \quad \text{(Eq. 4.8)}$$

The population ratio, $P_{(t)}^{ratio}$, is found by dividing projected population in each timestep by projected population in the initial timestep (Eq. 4.9). Observed population is then summed over all zones, z, (Eq. 4.10) allowing the calculation of study region population for each timestep by multiplying observed population by the population ratio (Eq. 4.11). The model of zonal population (section 4.5.3) spreads the change in population relative to the previous timestep so population change in each timestep, $P_{(t)}^{change}$, is calculated (Eq. 4.12).

$$P_{(t)}^{ratio} = P_{(t)}^{proj} / P_{(t=0)}^{proj} \quad \text{(Eq. 4.9)}$$

$$P^{obsrv} = \sum P_{(z)}^{obsrv} \quad \text{(Eq. 4.10)}$$

$$P_{(t)} = P_{(t)}^{ratio} \times P^{obsrv} \quad \text{(Eq. 4.11)}$$

$$P_{(t)}^{change} = P_{(t)} - P_{(t-1)} \quad \text{(Eq. 4.12)}$$

The Boolean parameters emp_driver and pop_driver (Figure 4.2) specify the projected data used by the model. Where both parameters are TRUE, employment and population are configured as described previously in this section. Where either value is FALSE, further calculations are needed to drive both employment and population from a single set of projected values.

Where only pop_driver is TRUE, it is assumed that there is a single sector of employment. Study region employment for each timestep can then be scaled from observed employment values using the population ratio (Eq. 4.9). Where only emp_driver is TRUE, the employment ratio calculation (Eq. 4.5) is modified to sum across all sectors, study region population for each timestep can then be scaled from observed population values using the all-sector employment ratio. Both of these approaches assume a fixed inverse activity ratio (Eq. 4.13) based on total observed population (Eq. 4.10) and total observed employment (Eq. 4.14).

$$Act^{inv} = P^{obsrv}/E^{obsrv} \quad \text{(Eq. 4.13)}$$

$$E^{obsrv} = \sum E^{obsrv}_{(z)(s)} \quad \text{(Eq. 4.14)}$$

4.5.2 Model of Zonal Employment

Employment change, $E^{change}_{(s)(t)}$, calculated in section 4.5.1 is spread across zones in the study region using input zonal employment attractors and weights, these are combined into a normalised employment mass term which sums to one across all zones for each sector.

The employment mass term, $M^{emp}_{(z)(s)}$, is used to spread positive employment change across zones for timesteps $t > 0$.

Input employment attractors are firstly summed over all zones (Att^{sum}) then normalised across all zones (Eq. 4.15) for each attractor, whilst input employment weights are firstly summed over all attractors (W^{sum}) then normalised across all attractors (Eq. 4.16) for each sector. Spatial employment attractors are used in conjunction with sector weights allowing each sector to be independently spatially driven. These are combined by copying attractors to all sectors then multiplying by weights to create a set of weighted employment attractors for all zones, attractors and sectors (Eq. 4.17). The employment mass term, $M^{emp}_{(z)(s)}$, is then calculated as the sum of weighted employment attractors over all attractors.

$$Att^{emp}_{(z)(e)} = Att^{emp}_{(z)(e)}/Att^{sum} \quad \text{(Eq. 4.15)}$$

$$W^{emp}_{(e)(s)} = W^{emp}_{(e)(s)}/W^{sum} \quad \text{(Eq. 4.16)}$$

$$wAtt^{emp}_{(z)(e)(s)} = Att^{emp}_{(z)(e)(s)} \times W^{emp}_{(e)(s)} \quad \text{(Eq. 4.17)}$$

A negative employment mass term, $M^{-emp}_{(z)(s)}$, is used to spread negative employment change (calculated in 4.5.1) across zones in the study region for timesteps $t > 0$. This uses the reciprocal of the employment mass calculated to spread positive employment (Eq. 4.18) on the basis that decline is most likely to occur in the least attractive areas for growth. The negative employment mass is calculated by firstly summing the reciprocal employment mass over all zones (rM^{sum}) then normalising across all zones (Eq. 4.19) for each sector.

$$rM_{(z)(s)}^{emp} = 1 - M_{(z)(s)}^{emp} \quad \text{(Eq. 4.18)}$$

$$M_{(z)(s)}^{-emp} = rM_{(z)(s)}^{emp} / rM^{sum} \quad \text{(Eq. 4.19)}$$

The spread of employment across zones for each sector is dependent upon whether employment change is positive or negative. Where employment change is positive, $E_{(s)(t)}^{change} > 0$, then disaggregation across zones uses the employment mass term (Eq. 4.20), where employment change is negative, $E_{(s)(t)}^{change} < 0$, then disaggregation across zones uses the negative employment mass term (Eq. 4.21). This initial spread is subject to corrections for negative employment (section 4.5.4) and land use capacity (section 4.5.6).

$$E_{(z)(s)(t)}^{change} = M_{(z)(s)}^{emp} \times E_{(s)(t)}^{change} \quad \text{(Eq. 4.20)}$$

$$E_{(z)(s)(t)}^{change} = M_{(z)(s)}^{-emp} \times E_{(s)(t)}^{change} \quad \text{(Eq. 4.21)}$$

4.5.3 Model of Zonal Population

The population mass term, $M_{(z)}^{pop}$, is used along with accessibility to employment to spatially disaggregate population change calculated in section 4.5.1. Input population attractors are firstly summed over all zones (Att^{sum}) then normalised across all zones (Eq. 4.22) for each attractor, whilst input population weights are firstly summed over all attractors (W^{sum}) then normalised across all attractors (Eq. 4.23). These are combined by multiplying attractors by weights to create a set of weighted population attractors for all zones and attractors (Eq. 4.24). The population mass term, $M_{(z)}^{pop}$, is then calculated as the sum of weighted population attractors over all attractors.

$$Att_{(z)(p)}^{pop} = Att_{(z)(p)}^{pop} / Att^{sum} \quad \text{(Eq. 4.22)}$$

$$W_{(p)}^{pop} = W_{(p)}^{pop} / W^{sum} \quad \text{(Eq. 4.23)}$$

$$wAtt_{(z)(p)}^{pop} = Att_{(z)(p)}^{pop} \times W_{(p)}^{pop} \quad \text{(Eq. 4.24)}$$

The population potential term, $Pot_{(z)(t)}^{pop}$, is used to spread population change across zones in the study region for timesteps $t > 0$. It combines the population mass term constructed from weighted attractors with all-sector employment and the accessibility matrix, $Acc_{(z)(z)}$, output from the TAM (Chapter 5).

For each origin zone, the product of all-sector employment in the destination zone, and the accessibility from origin to destination, is summed for all destinations (Eq. 4.25) to generate accessibility to employment which is multiplied by the population mass for the origin zone (Eq. 4.26). Population potential, $Pot_{(z)(t)}^{pop}$, is then summed over all zones (Pot^{sum}) and normalised across all zones (Eq. 4.27).

$$Acc_{(orig)(t)}^{emp} = \sum E_{(dest)(t)}^{total} \times Acc_{(orig)(dest)} \quad \text{(Eq. 4.25)}$$

$$Pot_{(orig)(t)}^{pop} = Acc_{(orig)(t)}^{emp} \times M_{(orig)}^{pop} \quad \text{(Eq. 4.26)}$$

$$Pot_{(z)(t)}^{pop} = Pot_{(z)(t)}^{pop} / Pot^{sum} \quad \text{(Eq. 4.27)}$$

A negative population potential term, $Pot_{(z)(t)}^{-pop}$, is used to spread negative population change (calculated in 4.5.1) across zones in the study region for timesteps $t > 0$. This uses the reciprocal of the population potential calculated to spread positive population (Eq. 4.28) on the basis that decline is most likely to occur in areas with the lowest accessibility to employment and the least attractiveness for growth. The negative population potential is calculated by firstly summing the reciprocal population potential over all zones ($rPot^{sum}$) then normalising across all zones (Eq. 4.29).

$$rPot_{(z)(t)}^{pop} = 1 - Pot_{(z)(t)}^{pop} \quad \text{(Eq. 4.28)}$$

$$Pot_{(z)(t)}^{-pop} = rPot_{(z)(t)}^{pop} \,/\, rPot^{sum} \quad \text{(Eq. 4.29)}$$

The spread of population across zones is dependent upon whether population change is positive or negative. Where population change is positive, $P_{(t)}^{change} > 0$, then disaggregation across zones uses the population potential term (Eq. 4.30), where population change is negative, $P_{(t)}^{change} < 0$, then disaggregation across zones uses the negative population potential term (Eq. 4.31). This initial spread is subject to corrections for negative population (section 4.5.4) and land use capacity (section 4.5.5).

$$P_{(z)(t)}^{change} = P_{(t)}^{change} \times Pot_{(z)(t)}^{pop} \quad \text{(Eq. 4.30)}$$

$$P_{(z)(t)}^{change} = P_{(t)}^{change} \times Pot_{(z)(t)}^{-pop} \quad \text{(Eq. 4.31)}$$

4.5.4 Negative Correction

In some circumstances the initial spread of negative change in employment sectors (Section 4.5.2 – Eq. 4.21) and population (Section 4.5.3 – Eq. 4.31) may result in some zones being allocated negative total sectoral employment and/or population; this is not physically possible and must therefore be corrected. The approach to these corrections taken in the UIMF SIM (figure 4.4 and table 4.2) gives back employment/population to zones with a deficit and removes employment/population from zones with a surplus. The SIM maintains values of total employment/population as well as changes relative to the previous timestep in order to make these corrections. Sets of temporary values are calculated from total employment/population in the previous timestep plus the initial change in the current timestep (Eq. 4.32) which are used in the correction process.

$$X_{(z)}^{temp} = X_{(z)(t-1)}^{total} + X_{(z)(t)}^{change} \quad \text{(Eq. 4.32)}$$

Figure 4.4 generalises the negative correction process and table 4.3 details specific operations for employment and population. A key difference between the processes is that negative employment must be corrected on a sector by sector basis to preserve total employment levels in each sector, meaning that the flow of operations shown in figure 4.4 must be iterated for multiple employment sectors.

The removal of employment/population from Z2 zones is equivalent to iterating the negative spread across zones in the models of employment (Section 4.5.2 – Eq. 4.21) and population (Section 4.5.3 – Eq. 4.31); the difference is that the models are only applied to zones in set Z2 which have a surplus hence the negative employment mass and negative population potential terms are normalised over set Z2 zones to distribute the summed deficit across them. After corrections have been made on the sets of temporary values the total employment/population (Eq. 4.33) and change in the current timestep (Eq. 4.34) are found and used to calculate the available area in each zone by updating the land use for employment (section 4.5.7) and population (section 4.5.8).

$$X_{(z)(t)}^{total} = X_{(z)}^{temp} \quad \text{(Eq. 4.33)}$$

$$X_{(z)(t)}^{change} = X_{(z)}^{temp} - X_{(z)(t-1)}^{total} \quad \text{(Eq. 4.34)}$$

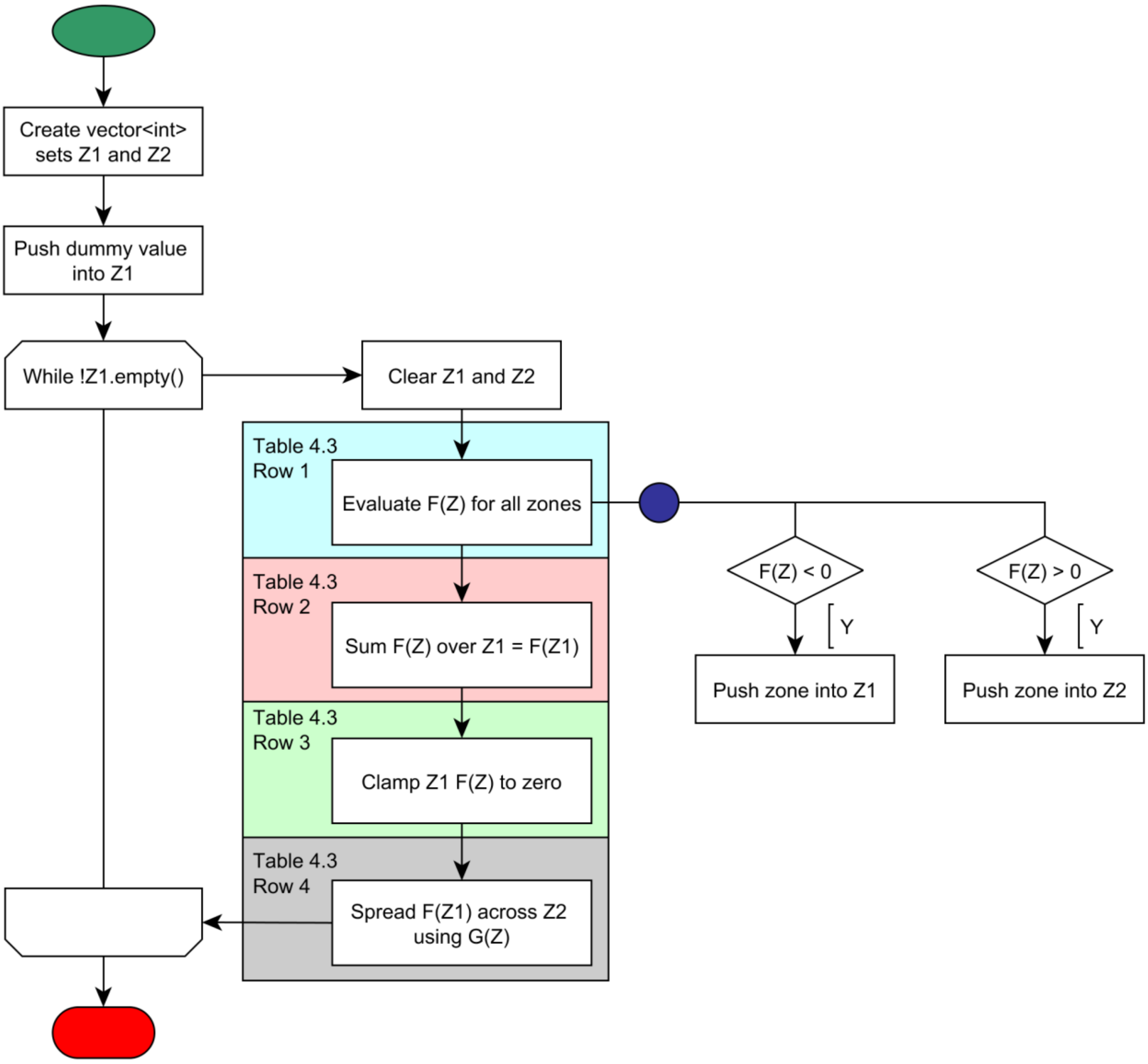

Figure 4.4: Overview of negative correction process.

Function – (see Figure N)	Employment Model	Population Model
Evaluate F(Z) for all zones	Total employment in sector	Total population
Sum F(Z) over Z1 = F(Z1)	Sum negative employment for sector in Z1 zones	Sum negative population in Z1 zones
Clamp Z1 F(Z) to zero	Clamp employment for sector to zero in Z1 zones	Clamp population to zero in Z1 zones
Spread F(Z1) across Z2 using G(Z)	G(Z) = Z2 normalised negative employment mass	G(Z) = Z2 normalised negative population potential

Table 4.3: Functions used in negative correction process (Figure 4.4).

4.5.5 Population Capacity Correction

In some circumstances the initial spread of positive change in population (Section 4.5.3 – Eq. 4.30) may result in some zones being allocated more population than can be housed at the input observed population density; this could be rectified by increasing the population density to meet demand within the available area but the approach taken in the UIMF SIM is to enforce capacity constraints within each zone and redistribute excess population where possible over zones with land availability. The UIMF SIM overflow output records whether there was enough land available in the study region to accommodate the projected increase in population for each timestep; where this was not the case, zonal population densities can be adjusted before rerunning the model.

Figure 4.5 provides an overview of the land use correction process which is structurally similar to the process of negative correction described in section 4.5.4. Excess population is evaluated (Figure 4.5 - *1 – F(Z)) by comparing the initial spread of population change with the maximum change possible within each zone in terms of population density and remaining free area within the zone (Eq. 4.35). Zones with spare capacity are pushed into set Z2 whilst zones which exceed capacity are pushed into set Z1 and excess population is summed across the set. Population change in Z1 zones is set to the maximum possible for each zone (Figure 4.5 - *2) at current population density within available land.

$$P_{(z)}^{excess} = P_{(z)(t)}^{change} - \left(D_{(z)}^{pop} \times A_{(z)}^{free}\right) \quad \text{(Eq. 4.35)}$$

The redistribution of excess population across Z2 zones is equivalent to iterating the positive spread in the model of zonal population (Section 4.5.3 – Eq. 4.30); the difference is that the model is only applied to zones in set Z2 which have spare capacity hence the population potential term (Figure 4.5 - *3 – G(Z)) is normalised over set Z2 zones to distribute the summed excess population across them.

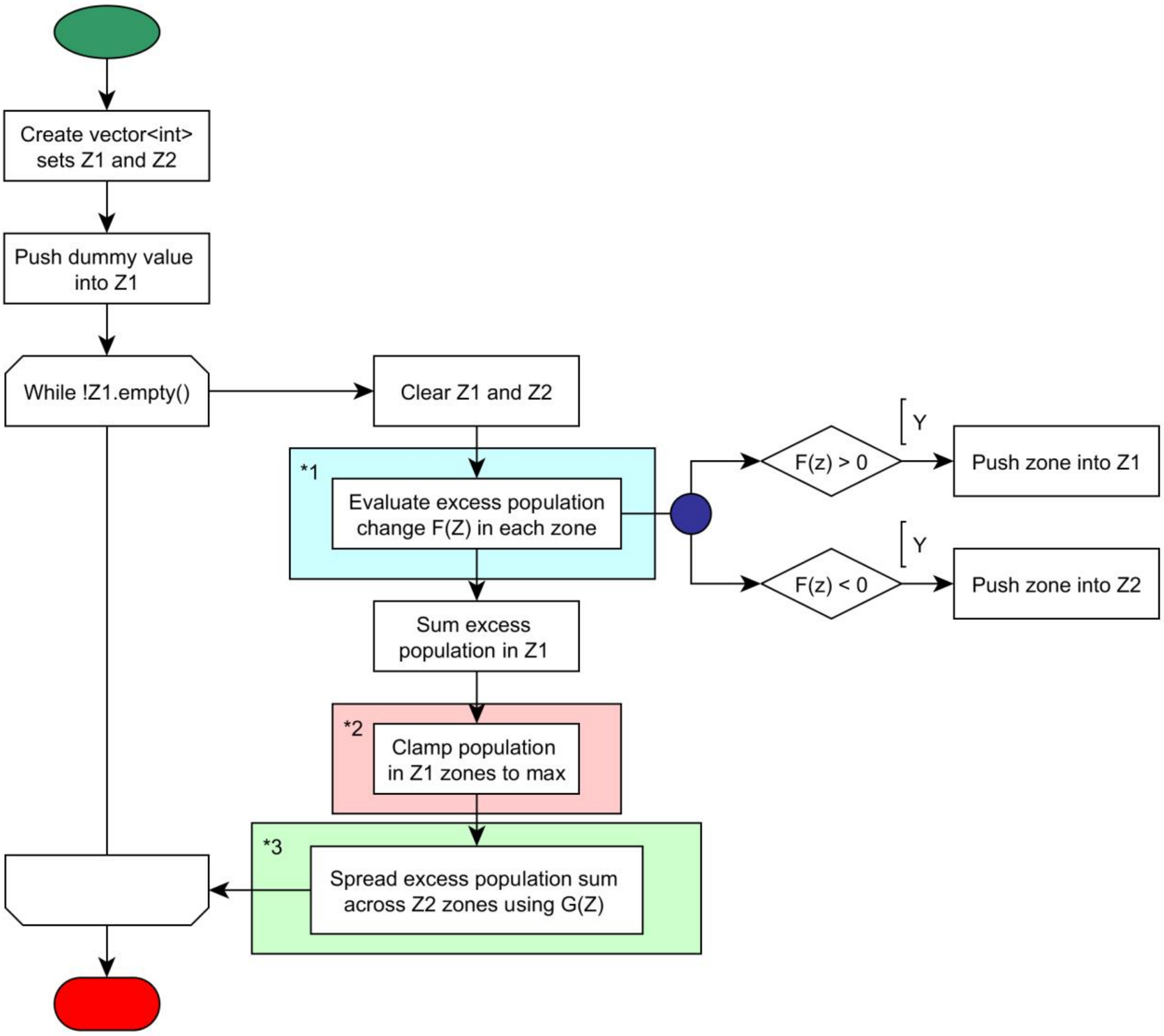

Figure 4.5: Population Capacity Correction.

4.5.6 Employment Capacity Correction

In some circumstances the initial spread of positive change in employment (Section 4.5.2 – Eq. 4.20) may result in some zones being allocated more employment than can be housed at the input observed employment density; this could be rectified by increasing the employment density to meet demand within the available area but the approach taken in the UIMF SIM is to enforce capacity constraints within each zone and redistribute excess employment where possible over zones with land availability. The UIMF SIM overflow output records whether there was enough land available in the study region to accommodate the projected increase in employment for each timestep; where this was not the case, zonal employment densities can be adjusted before rerunning the model.

Figure 4.6 provides an overview of the employment capacity correction process which is similar to population capacity correction (Section 4.5.5) but is more complex due to the need to handle employment change over multiple sectors. Key details corresponding to the shaded and numbered boxes in Figure 4.6 are as follows:

1. Employment is summed across sectors with positive employment change within each zone; this is because sectors with negative employment change have already given land back to the zone in proportion to employment density and the amount of negative change. Excess employment, (F(Z)) in figure 4.6, is evaluated by comparing the initial spread of employment change in positive sectors with the maximum change possible within each zone in terms of employment density and remaining free area within the zone (Eq. 4.36). Zones which exceed capacity are pushed into set Z1 whilst zones with spare capacity are pushed into set Z2.

$$E_{(z)}^{excess} = E_{(z)(+s)(t)}^{change} - \left(D_{(z)}^{emp} \times A_{(z)}^{free}\right) \quad \text{(Eq. 4.36)}$$

2. In Z1 zones the excess employment is as calculated in Eq. 4.36 and the maximum employment possible is the product of employment density and area remaining in the zone. In each zone the employment change in positive sectors is summed and normalised across sectors, G(Z) in figure 4.6, in support of steps 3 and 4.

3. Employment change in each Z1 zone is recalculated using the maximum employment value and the normalisation across sectors, G(Z) in figure 4.6, calculated in step 2. This ensures that the employment in each sector is in proportion to the initial spread.

4. Excess employment change in each zone is split into sectors using the normalisation across sectors calculated in step 2. This allows the excess to be redistributed across Z2 zones on a sector by sector basis, equivalent to iterating the positive spread in the model of zonal employment (Section 4.5.2 – Eq. 4.20); the difference is that the model is only applied to zones in set Z2 which have spare capacity hence the employment mass term (Figure 4.6 – H(Z)) is normalised over set Z2 zones to distribute the summed excess employment across them for each sector.

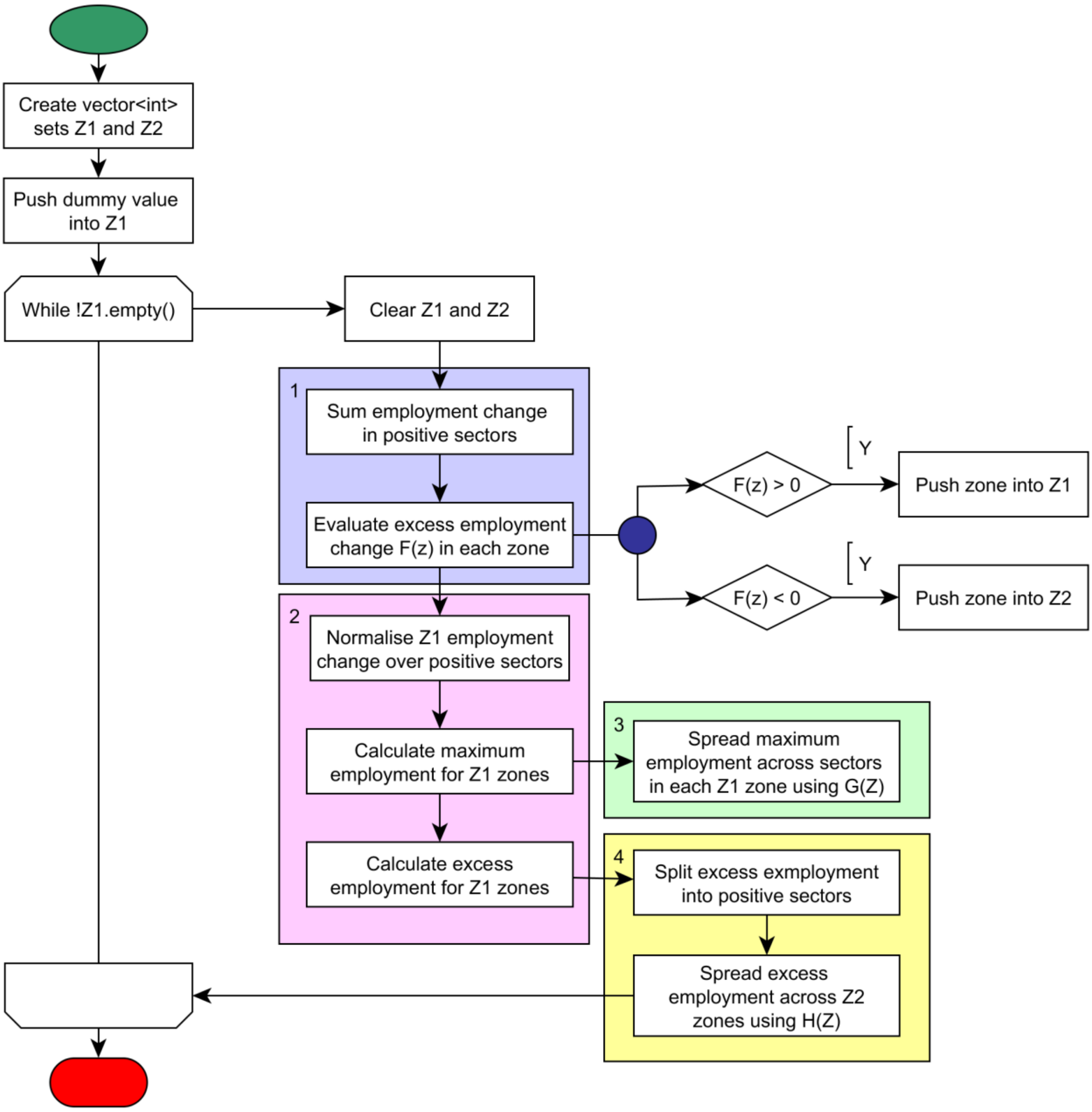

Figure 4.6: Employment Capacity Correction.

4.5.7 Update Employment Land Use

Having corrected the spread of employment change for the current timestep the free area within each zone must be updated. After negative correction the area from contracting employment sectors (Eq. 4.37) is added back to zones. After capacity correction the area used by positive employment change is removed from each zone (Eq. 4.38).

$$A_{(z)}^{free} -= E_{(z)(s)(t)}^{-change} / D_{(z)}^{emp} \quad \text{(Eq. 4.37)}$$

$$A_{(z)}^{free} -= E_{(z)(t)}^{+change} / D_{(z)}^{emp} \quad \text{(Eq. 4.38)}$$

4.5.8 Update Population Land Use

Having corrected the change in population change for the current timestep the free area within each zone must be updated. The area from contracting population is added back to zones (Eq. 4.39). The area used by positive population change is then removed from each zone (Eq. 4.40).

$$A_{(z)}^{free} -= P_{(z)(t)}^{-change} / D_{(z)}^{pop} \quad \text{(Eq. 4.39)}$$

$$A_{(z)}^{free} -= P_{(z)(t)}^{+change} / D_{(z)}^{pop} \quad \text{(Eq. 4.40)}$$

4.6 Summary

This chapter has presented the Spatial Interaction Model (SIM) implemented in the Urban Integrated Modelling Framework (UIMF). Section 4.2 discussed theory and justified the simplified approach taken. Section 4.3 described the development of this model from the reference implementation in the Urban Integrated Assessment Framework (UIAF) and identified key differences between them. Section 4.4 provided a high-level overview of the SIM and its operation within the software environment of the UIMF, and Section 4.5 gave a detailed description of the SIM implementation. The following chapter details the Transport Accessibility Model (TAM) which provides the SIM with an origin-destination matrix of transport accessibility.

Chapter 5: Transport Accessibility Model

5.1 Introduction

This chapter presents the Transport Accessibility Model (TAM) implemented in the Urban Integrated Modelling Framework (UIMF). Section 5.2 provides theoretical underpinnings, compares the TAM with similar models and justifies the simplified approach taken. Section 5.3 describes the development of this model from the reference implementation in the Urban Integrated Assessment Framework (UIAF) and highlights key differences. Section 5.4 gives a high-level overview of the model and its operation within the software environment of the UIMF, while Sections 5.5 to 5.9 provide detailed descriptions of each stage of the TAM implementation.

5.2 Context

Accessibility is a key driver of urban growth and locational choice, as described in 3.4.2 where aggregate utility models are based upon accessibility, which is defined as the opportunity for spatial interactions at any given location. The potential accessibility of any given location can be found by summing activities across destinations and weighting by an inverse function of the distance, time, or generalised cost of reaching them. The Spatial Interaction Model (SIM – Chapter 4) is in fact a simple utility model which uses accessibility to employment as a spatial driver of residential location choice.

Aggregate approaches to accessibility such as the TAM are based on spatial zones which are approximated by geometric centroids to from trip origins and destinations. The analysis of transport networks via which these trips are made is based on the computation of 'shortest' or 'least-cost' paths between each pair of origin and destination nodes. Factors relating to time and monetary costs are attributed to network edges which forms the basis of finding 'least-cost' network routes across weighted edges using Dijkstra's algorithm.

The UIMF retains the following key assumptions of the UIAF Accessibility Assessment Framework (AAF):

- Trips between origins and destinations are via a single mode of transport
- Congestion is not modelled therefore network edges are assumed to have infinite capacity

The first of these simplifications can be addressed by further model development to allow interchanges between transport modes whilst applying costs to network edges which are relevant to each mode. The second assumption can be addressed to an extent using traffic counts and public transport service reliability as a measure of observed congestion which would improve the representation of trip costs at the expense of significantly increased data requirements. The problem with this approach is that when networks are modified to simulate the construction of new routes there is no observed congestion data for new network edges or the impact of their construction on the congestion of other routes. A full macro approach such as the four-step transport model or microsimulation of traffic such as that provided by MATSim or SUMO, could be used to address this problem but exhaustive data requirements and long computation times can render such models unsuitable in the context of rapid assessment and can prevent their uptake by non-expert users.

5.3 Model Development

The UIAF AAF was developed using Visual Basic for Applications (VBA) as an add-in for ESRI's ArcGIS applying the Network Analyst extension to compute shortest routes using Dijkstra's algorithm. The UIMF TAM was developed to match the capabilities of this framework in terms of network processing and the generation of origin-destination matrices of generalised cost for accessibility analysis. An early prototype using Python and the NetworkX library was developed using the process description provided by Ford *et al* (2015) as summarised in Figure 5.1.

Initial development focussed on connecting zone centroids to their nearest network access points, building the networks and storing them in PostgreSQL tables before computing shortest paths using NetworkX. Development of the TAM continued by incorporating generalised cost parameters such as public transport fares and private vehicle occupancy, making the model portable between geographical regions using a spatial reference system identifier and adding desirable functionality based upon discussions with the UIAF AAF development team. These features include:

- The automated generation of interchange edges between public transport stops for a single mode of transport
- A flexible charging system based on an input charge zone polygon for private vehicles
- Combining costs from multiple modes of transport for a given scenario
- Converting costs to accessibility using a deterrence function

The final stage of development was optimisation; network building was identified as a bottleneck during initial testing along with the search for shortest paths using Dijkstra's algorithm. This led to the separation of network processing and cost models, and the use of

C++ and the igraph library for pathfinding as opposed to Python and the NetworkX package, resulting in significant performance gains. A comparison of features between the UIAF AAF and the UIMF TAM is shown in table 5.1.

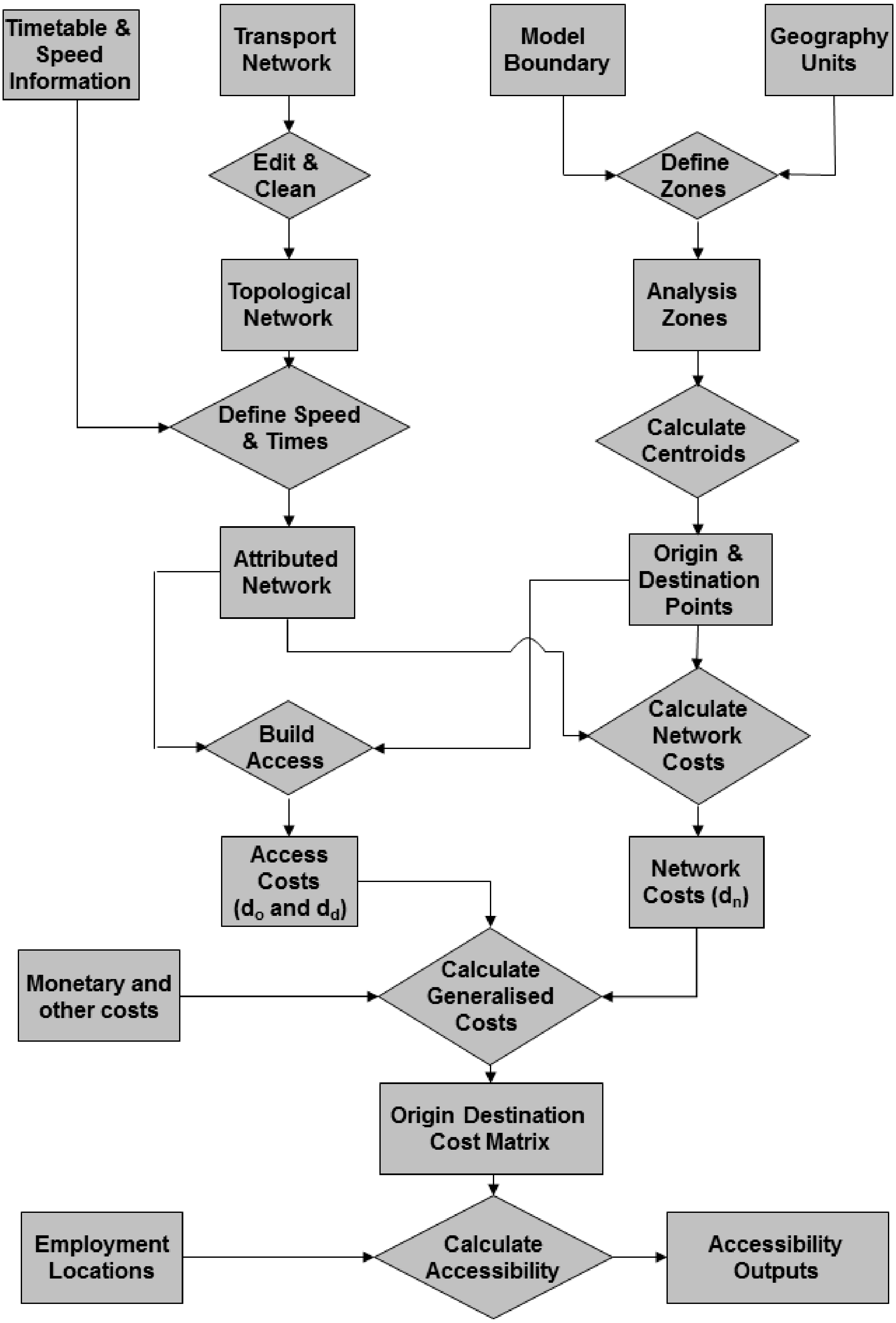

Figure 5.1: Computational Framework for Transport Accessibility Analysis (Ford et al, 2015)

Feature	UIAF AAF	UIMF TAM
Interchange between public transport stops	Described for London case study (Ford et al, 2015)	Generated automatically given a maximum interchange distance value.
Zone-based charging	Described for London case study (Ford et al, 2015)	Flexible system based on geometry testing against network edges and zone centroids
Dijkstra's algorithm performance	Unknown	Optimised using C/C++ and igraph for high performance
Cost matrix aggregation across transport modes	No	Yes
Link to spatial interaction model	Manual loose coupling	Integrated coupling via UIMF model group chaining
Metadata	No	Yes
Graphical User Interface	Yes	No
Open source	No	Yes

Table 5.1: Feature Comparison between UIAF AAF and UIMF TAM

5.4 Model Overview

As shown in Figure 5.2, the TAM comprises five sub-models allowing both private and public transport networks to be assessed in terms of origin-destination trips, generalised cost and accessibility. The private network processing and cost models (5.5 and 5.6) can be used for cars and bicycles, whilst the public network processing and cost models (5.7 and 5.8) can be used for a variety of public transport modes such as bus or train. This flexibility allows the TAM to be applied to different urban areas with different transport infrastructure where the generalised costs of travel for each transport mode are combined to generate a single origin-destination matrix of transport accessibility (5.9). This accessibility matrix is used by the Spatial Interaction Model (SIM – Chapter 4) in combination with zonal employment as a spatial driver of residential location choice requiring that the order of zones is consistent across models as indicated by the dotted line.

For both private and public modes of transport, the process of generating trip costs is split into two stages due to the execution time of network processing models when working with large network datasets. It is only necessary to run the network processing models when simulating physical changes such as the construction of new roads or stations, this enables the impact of changes in parameters to the cost models to be explored more efficiently.

Network processing models connect zone centroids to their nearest network access points and operate directly on data stored in PostgreSQL tables using PostGIS for spatial operations such as intersection testing. These PostGIS functions are indicated in the text in italics, for example, $ST\_Within()$. The Python package, NetworkX is used to construct modified transport networks geometrically from tables of network nodes (points) and edges (linestrings). The nx_pgnet library developed at Newcastle University is then used to write processed networks to PostgreSQL tables with a specific format supporting graph traversal and pathfinding; the nx_pgnet library is a network database schema and Application Programming Interface (API) for storing networks within a relational database with an interface to python and the NetworkX network format.

Network cost models apply cost parameters to set edge weights and carry out pathfinding making use of the computational flexibility of the UIMF by using C++ and the igraph library to optimise the computation of ‘least-cost’ paths between origins and destinations using Dijkstra’s algorithm. All sub-models of the TAM write data and metadata back to the database such that the metadata recorded by the final sub-model (5.9) contains network processing and cost model metadata for each mode of transport in the study region.

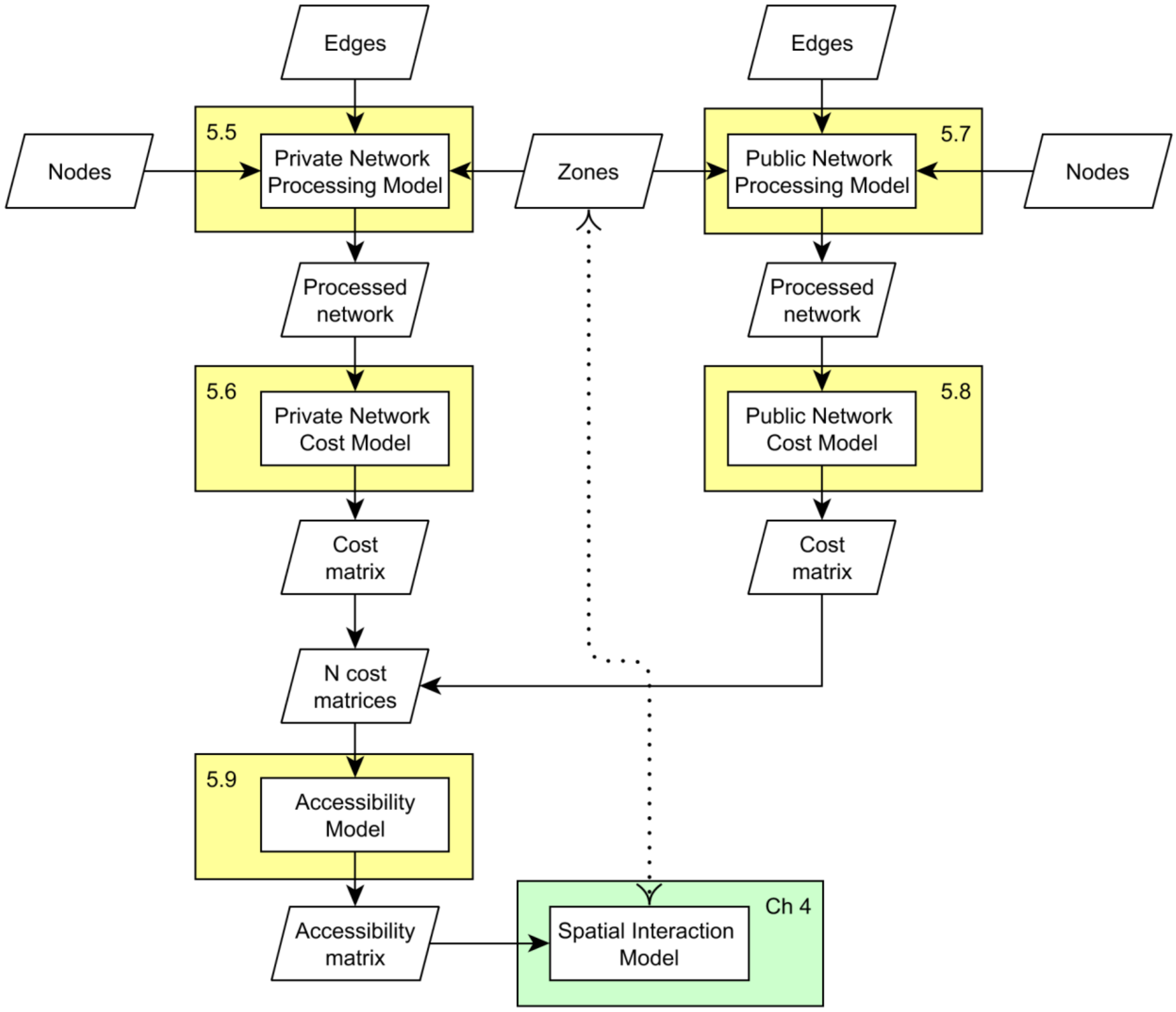

Figure 5.2: Transport Accessibility Model Overview

5.5 Private Network Processing Model

As shown in Figure 5.3, the private network processing model operates on the geometries of an input transport network, and a collection of zones consistent with the spatial interaction model (Chapter 4). The input network comprises points for ‘centroid’ nodes, N, and linestrings for ‘route’ edges, E, where nodes are point locations which serve as origins and destinations of trips between zones, and edges are roads in the network. The polygon zones, Z, define the spatial extent of each zone and are used to optimise the search for nearest ‘route’ edges when modifying the network.

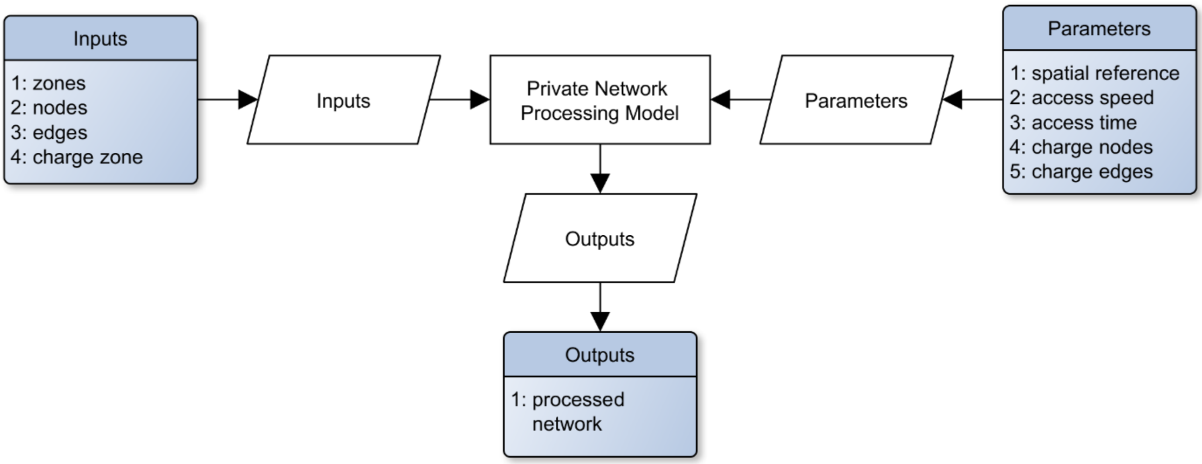

Figure 5.3: Private Network Processing Model Interface

The overall function of the model is to generate new ‘access’ edges between zone centroids and nearest ‘route’ edges thus enabling network traversal from origins to destinations where connections permit. The time taken to travel along ‘access’ edges from ‘centroid’ nodes to the nearest ‘route’ edge is calculated using the access speed parameter.

The optional input charge zone is a geometrically defined region within which charges apply; this acts in conjunction with the parameters charge centroids and charge edges to perform tests on input geometry in support of flexible charging in the private transport cost model (5.6). Enabling charge centroids tests whether ‘centroid’ nodes are within the charge zone whereas enabling charge edges tests whether ‘route’ edges and ‘access’ edges cross the charge zone boundary.

The model uses the spatial reference parameter to add new ‘access’ edges to the network and outputs a processed network which is used in the private transport cost model (5.6). The flow of operations in the model are described in the following subsections as indicated in Figure 5.4.

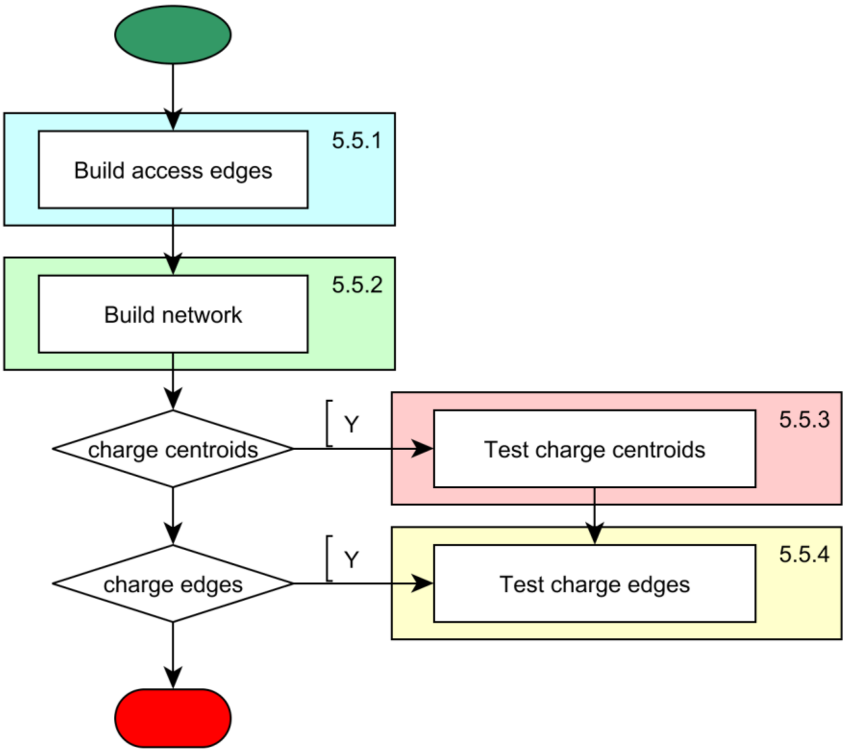

Figure 5.4: Private Network Processing Model Overview

5.5.1 Build Access Edges

This function iterates over all zones and carries out the following operations as depicted in Figure 5.5:

- **Finding the nearest edge**
  For each 'centroid' node, the search for the nearest 'route' edge is optimised using the function $ST\_Intersects()$ leaving only 'route' edges which touch, overlap, or are within the zone polygon. The nearest 'route' edge is found by applying the function $ST\_Distance()$ to each 'route' edge along with the 'centroid' node of the zone, ordering results by the value returned.
- **Finding the nearest point on the nearest edge**
  The point on this nearest 'route' edge which is nearest to the 'centroid' node is first found in terms of the fractional edge length returned by the function $ST\_Line\_Locate\_Point()$. The function $ST\_Line\_Interpolate\_Point()$ is then used to locate the point in terms of the fractional edge length and the linestring representing the nearest 'route' edge.

- **Adding access edge**

  New ‘access’ edges are generated between ‘centroid’ nodes and the nearest point on the nearest ‘route’ edge using the function $ST\_Makeline()$. These linestrings are measured using the function $ST\_Distance()$ which, in conjunction with the access speed parameter, allows the time taken to access a nearest ‘route’ edge to be calculated.

- **Splitting nearest edge**

  The nearest ‘route’ edge is then split into two sub-edges, A and B, at the nearest point to the centroid. The function $ST\_Line\_Substring()$ is used to extract a substring of the nearest ‘route’ edge linestring where sub-edge A runs from 0 to the fractional edge length at the point nearest to the ‘centroid’ node, and sub-edge B runs from that point to 1. The length of each sub-edge is then found using the function $ST\_Length()$ and the split ‘route’ edge is removed from the network.

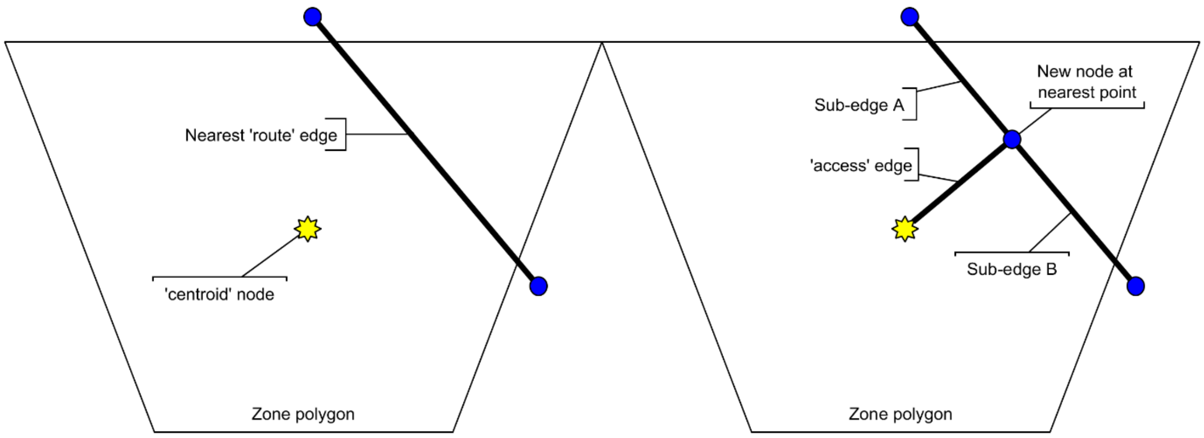

Figure 5.5: Private Access Edges

5.5.2 Build Network

NetworkX is used to construct the modified transport network geometrically from a table of nodes containing ‘centroid’ nodes, and a table of edges containing ‘route’ edges and ‘access’ edges. The nx_pgnet library is used to write the processed network to PostgreSQL tables with a specific format supporting graph traversal and pathfinding; network nodes have a unique identifier attribute, whilst network edges have a ‘from’ node attribute and a ‘to’ node attribute which link to the node unique identifiers.

5.5.3 Test Charge Centroids

When enabled by the charge centroids parameter, this function tests whether each ‘centroid’ node is within the charge zone using the function $ST\_Within()$ and the input charge zone polygon. The results of these tests are retained for use in the private transport cost model (5.6).

5.5.4 Test Charge Edges

When enabled by the charge edges parameter, this function tests whether each 'route' edge and 'access' edge cross the input charge zone boundary.

Firstly, the function tests whether each 'route' node and 'centroid' node is within the charge zone using the function $ST\_Within()$ and the input charge zone polygon. Then, using the nx_pgnet table format described in subsection 5.5.2, results for the 'from' node and 'to' node of each edge are tested using an Exclusive OR (XOR) gate to determine whether the edge starts and ends on different sides of the input charge zone boundary. The results of these tests are retained for use in the private transport cost model (5.6).

5.6 Private Network Cost Model

As shown in Figure 5.6, the private network cost model operates on the processed transport network output from the private network processing model (5.5). The parameters walking weight, occupancy, value of time and parking charge are all used in the calculation of generalised cost along with parameters for fuel and non-fuel costs which define vehicle operating costs. The Boolean parameters charge centroids and charge edges are used to enable a flexible charging system based upon the location of trip origins, destinations and routes with respect to a charging zone. This functionality uses a range of further input cost parameters and is based upon geometry tests performed in the private network processing model (5.5).

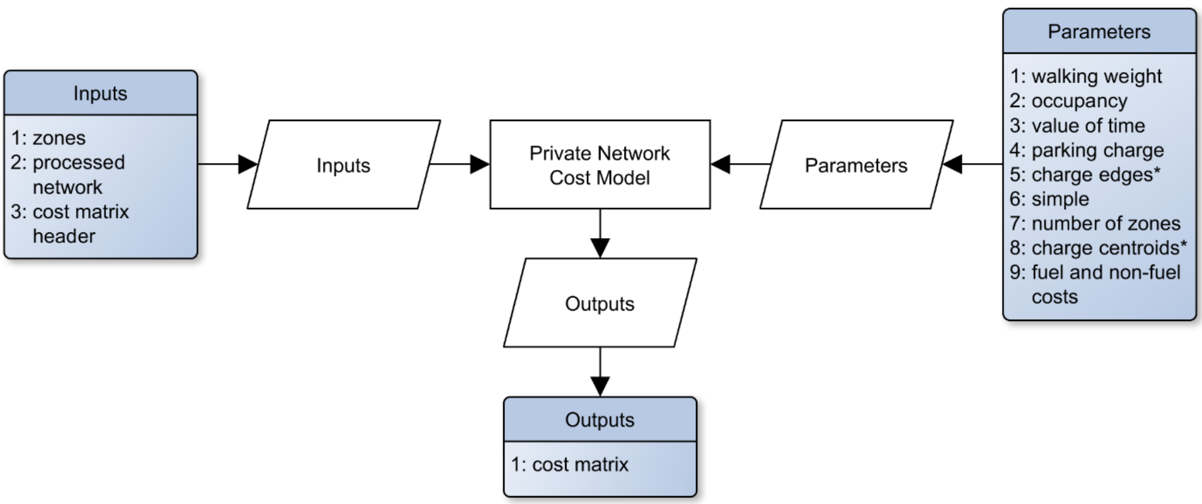

Figure 5.6: Private Network Cost Model Interface

The overall function of the model is to apply input cost parameters to the calculation of weights for each network edge. These weighted edges are then used to compute least cost paths between origin and destination nodes which are combined with other charging parameters to generate an origin-destination matrix of trip costs. The Boolean parameter simple is used to run a barebones version of the model by disregarding most parameters in the calculation of edge weights. This can be used to for cycle routes or to assess trips via any mode of transport in terms of time or distance only, providing functionality in use cases with limited data availability. The flow of operations in the model are described in the following subsections as indicated in Figure 5.7.

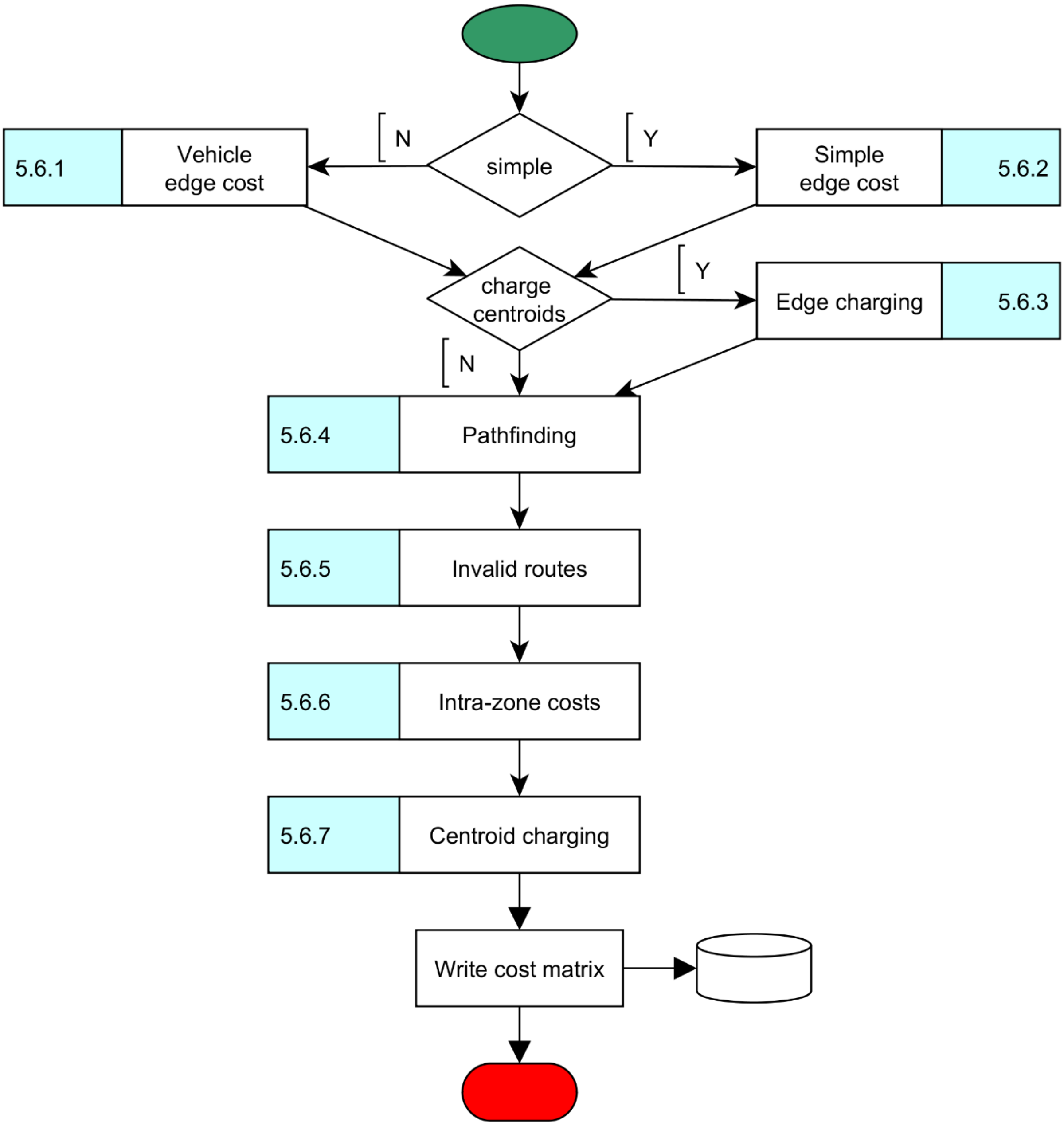

Figure 5.7: Private Network Cost Model Overview

The generalised cost for private transport networks is calculated as shown in Equation 5.1.

$$C_{pvt} = (V_{wk} \times A) + T + D * \frac{VOC}{occ*VOT} + \frac{PC}{occ*VOT} \quad \text{(Eq.5.1)}$$

Where V_{wk} is the walking disincentive weight; A is the network access time; T is the journey time; D is the journey distance; VOC is the vehicle operating cost; occ is the vehicle occupancy; VOT is the value of time; and, PC are parking and other charges.

Vehicle operating costs are calculated as shown in Equation 5.2.

$$VOC = L + C \quad \text{(Eq.5.2)}$$

Where L are fuel costs as shown in Equation 5.3, and C are non-fuel costs as shown in Equation 5.4.

$$L = \frac{a}{V} + b + c * V + d * V^2 \quad \text{(Eq.5.3)}$$

Where a, b, c, d are parameters defined for each vehicle category; and, V is the average speed.

$$C = a + \frac{b}{V} \quad \text{(Eq.5.4)}$$

Where a is a distance related costs parameter including maintenance and depreciation; b is a vehicle capital saving parameter; and, V is the average speed. Parameter b is only relevant for commercial vehicles so for non-commercial vehicles the non-fuel operating costs simplify to the parameter a as in Equation 5.6.

$$C = a \quad \text{(Eq. 5.6)}$$

5.6.1 Vehicle Edge Cost

The cost of 'access' edges, C_{pvt}^{access}, is calculated as shown in Equation 5.7 whilst the cost of 'route' edges, C_{pvt}^{route}, is calculated as shown in Equation 5.8 where vehicle operating cost parameters are applied as in Equation 5.9.

$$C_{pvt}^{access} = V_{wk} * A + \frac{PC/2}{occ*VOT} \quad \text{(Eq.5.7)}$$

$$C_{pvt}^{route} = T + D * \frac{VOC}{occ*VOT} \quad \text{(Eq.5.8)}$$

$$VOC = \frac{a}{V} + b + c * V + d * V^2 + C \quad \text{(Eq.5.9)}$$

Where D (length), V (speed), A and T (time) were computed in the private network processing model (5.5) and stored as attributes of network edges.

5.6.2 Simple Edge Cost

The cost of 'access' edges, C_{simple}^{access}, is calculated as shown in Equation 5.10 whilst the cost of 'route' edges, C_{simple}^{route}, is calculated as shown in equation 5.11.

$$C_{simple}^{access} = V_{wk} * A \quad \text{(Eq.5.10)}$$

$$C_{simple}^{route} = T \quad \text{(Eq.5.11)}$$

Where A and T (time) were computed in the private network processing model (5.5) and stored as attributes of network edges.

5.6.3 Edge Charging

Where edge charging is enabled, the input $charge$ cost parameter is added to the existing cost of 'access' edges and 'route' edges which cross the charge zone boundary as shown in Equation 5.12. This is based upon geometry tests performed in the private network processing model (5.5).

$$C^{charge} = C + \frac{charge}{occ * VOT} \quad \text{(Eq.5.12)}$$

5.6.4 Pathfinding

Network nodes, edges and costs are mapped to arguments of Dijkstra's shortest paths function implemented in the igraph library as shown in Figure 5.8:

1. **Graph construction:** An undirected empty graph with vertices equal to the number of nodes is created before an array of interleaved edge identities is allocated whose size is twice the number of edges. Iterating over all network edges, node identities of 'from' nodes and 'to' nodes are added to the interleaved array which is indexed by an incremental count variable.
2. **Origins and Destinations:** The order of zones in the origin-destination matrix of trip costs must be consistent with that used by all other models in the framework. A simple 'Node' data structure with an integer identity and a string label is used to store 'centroid' nodes and sort ascending by label to enforce zone ordering.
3. **Edge Weights:** The generalised cost calculated for each edge in the network (5.6.1) (5.6.2) (5.6.3) is used as a weight.

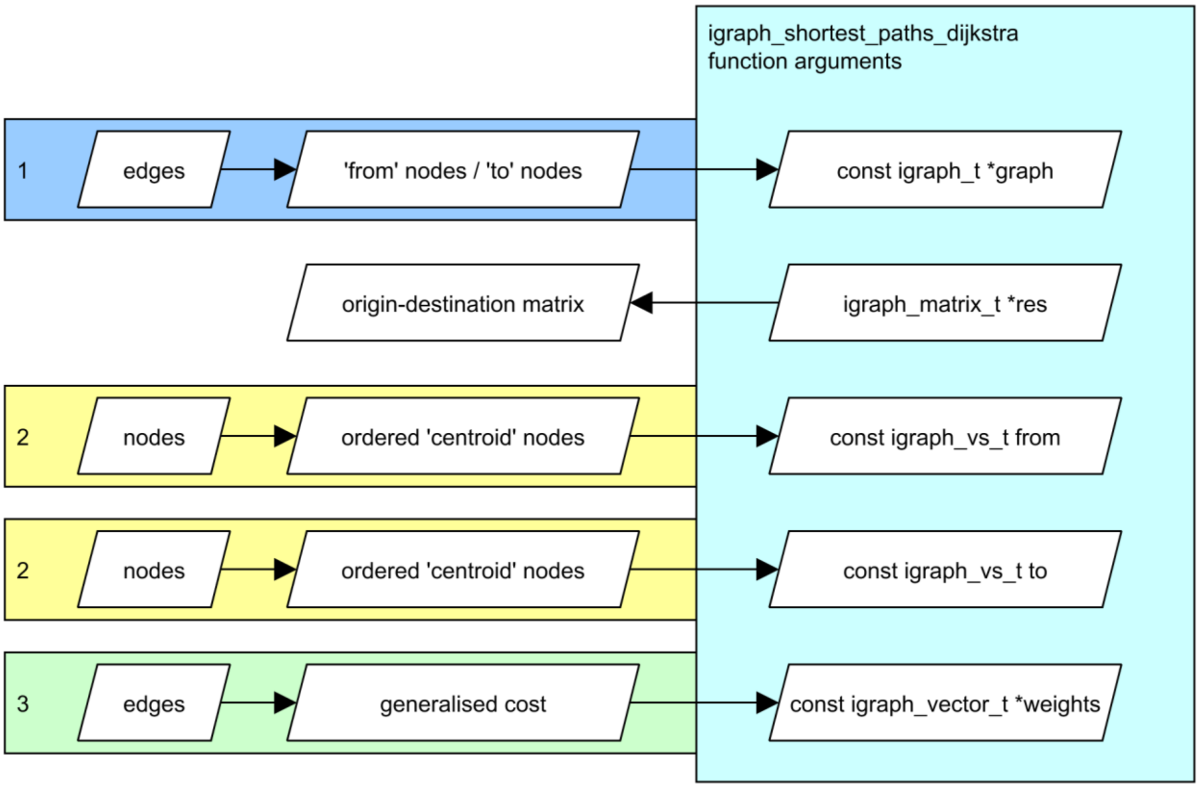

Figure 5.8: Mapping Private Transport Networks to Dijkstra's Algorithm

5.6.5 Invalid Routes

Iterating over all rows and columns of the output cost matrix, cells with the value of $igraph\_infinity$ indicating that no path could be found for a given origin-destination pair are set to the no data value specified in the cost matrix header.

5.6.6 Intra-zone Costs

The intra-zone trip cost (where origin = destination) for any given zone is set at two thirds of the minimum inter-zone trip cost for that zone (Ford et al, 2015) which is found for each origin by searching all destinations in the cost matrix.

5.6.7 Centroid Charging

Centroid charging is based upon geometry tests performed in the private network processing model (5.5) to establish whether 'centroid' nodes are within the charge zone boundary as indicated in Table 5.2. This functionality can be used in combination with edge charging, for instance, to apply charges to edges which cross the charge zone boundary and then apply discounts to origins within the charge zone.

centroid	charge	charge	charge	charge
origin	0	0	1	1
destination	0	1	0	1

Table 5.2: Centroid Charging Combinations

Iterating over all rows and columns of the output cost matrix, all four combinations of origin and destination zone status are accounted for when adding provided charges to values currently held in the cost matrix. Note that the provided monetary charge values are converted to generalised cost as shown in Equation 5.13.

$$Off = \frac{Off}{occ * VOT} \quad \text{(Eq. 5.13)}$$

5.7 Public Network Processing Model

As shown in Figure 5.9, the public network processing model operates on the geometries of an input transport network, and a collection of zones consistent with the spatial interaction model (see Chapter 4). The input network comprises points for 'stop' nodes, N, and linestrings for 'route' edges, E, where nodes are locations of public transport stops, and edges are routes between stops in the network. The polygon zones, Z, define the spatial extent of each zone and are used to calculate point centroids which serve as origins and destinations of trips between zones.

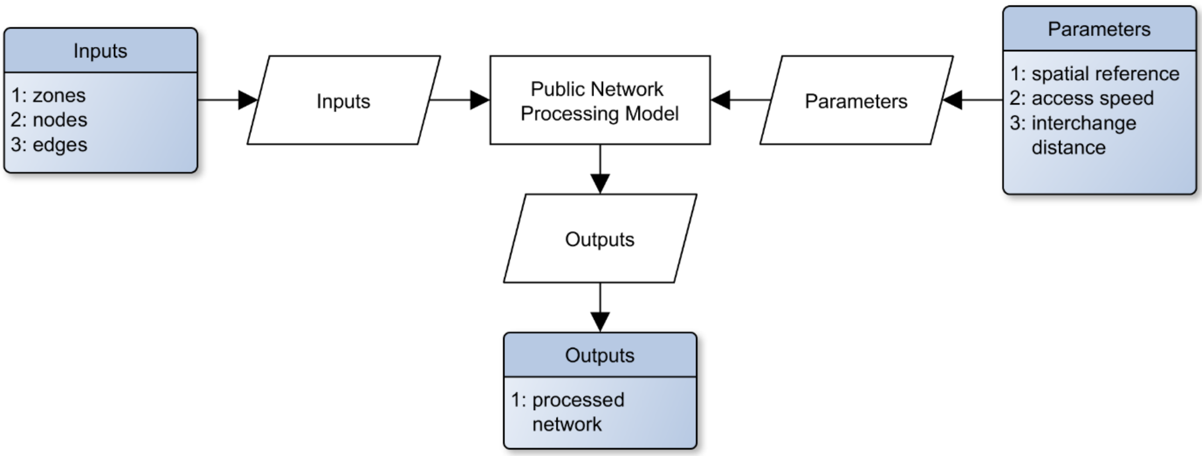

Figure 5.9: Public Network Processing Model Interface

The overall function of the model is to generate new ‘access’ edges between zone centroids and nearest public transport stops thus enabling network traversal from origins to destinations where connections permit. The access speed parameter is used in the calculation of time taken to reach nearest stops. The model also generates new ‘interchange’ edges between stops within a range specified by the interchange distance parameter. This allows connections to be made between different routes at stations etc. The model uses the spatial reference parameter to add these new edges to the network and outputs a processed network which is used in the public transport cost model (5.8). The flow of operations in the model are described in the following subsections as indicated in figure 5.10.

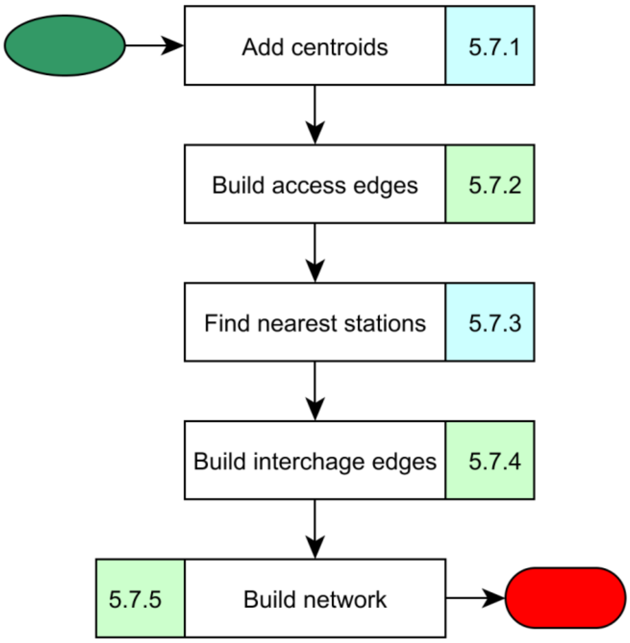

Figure 5.10: Public Network Processing Model Overview

5.7.1 Add Centroids

The centroid point locations which approximate input zones by acting as origins and destinations of trips are found by applying the function $ST\_Centroid()$ to the polygons of input zones. These new points are identified as ‘centroid’ nodes and are added to the input table holding transport stops identified as ‘stop’ nodes.

5.7.2 Build Access Edges

The public transport cost model (5.8) calculates the cost of trips between all pairs of origin and destination zones in the study region.

In the case of zones containing multiple transport stops, this cost is averaged for all routes accessed via all transport stops, this requires that 'access' edges are generated between the 'centroid' node and all 'stop' nodes within the zone. The condition that 'stop' nodes are in the same zone as the 'centroid' node is enforced by the function $ST\_Within()$ using the zone polygon from which the centroid was calculated. In the case of zones which contain no transport stops, the nearest stop outside of the zone is found.

New 'access' edges are generated between 'centroid' nodes and 'stop' nodes using the function $ST\_Makeline()$. These linestrings are measured using the function $ST\_Distance()$ which, in conjunction with the access speed parameter, allows the time taken to access a nearest station to be calculated. Both methods of generating 'access' edges for public transport networks are depicted in figure 5.11.

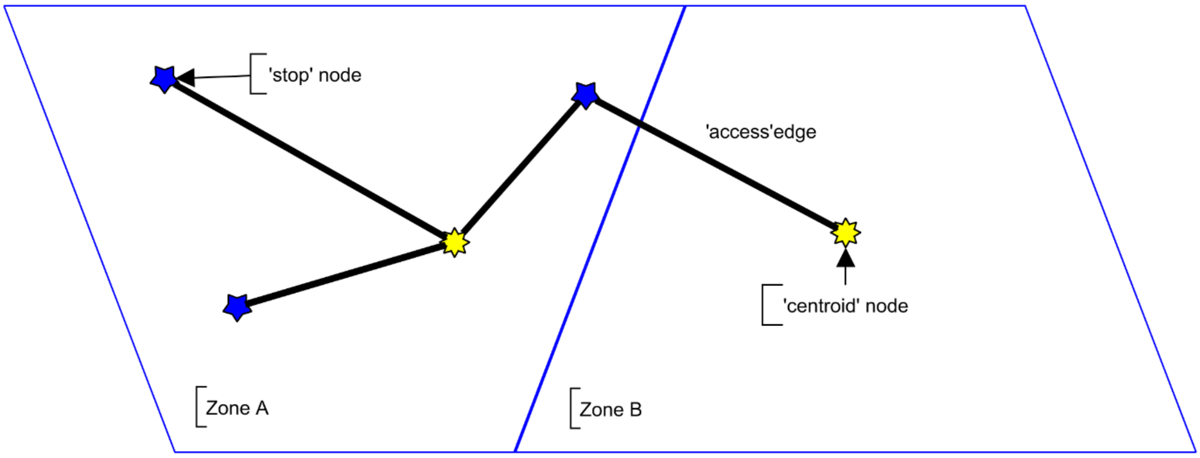

Figure 5.11: Public Access Edges

5.7.3 Find Nearest Stations

The public transport cost model (5.8) eliminates trips between origins and destinations which have the same nearest transport stop, this prevents attribution to the transport network of nonsensical trips containing only 'access' edges routed via a 'stop' node. In order to eliminate these trips, the nearest 'stop' node to each 'centroid' node is recorded at this stage.

5.7.4 Build Interchange Links

To allow connections to be made at stations the model generates new 'interchange' edges between transport stops within a radial distance specified by the interchange distance parameter. New 'interchange' edges are created between 'stop' nodes and other 'stop' nodes using the function $ST\_Makeline()$. These linestrings are measured using the function $ST\_Distance()$ which, in conjunction with the walking speed parameter, allows the time taken to walk between transport stops to be calculated. The condition that 'stop' nodes are within the specified range of one another is enforced using the function $ST\_Within()$. The method of building interchange edges for public transport networks is depicted in Figure 5.12.

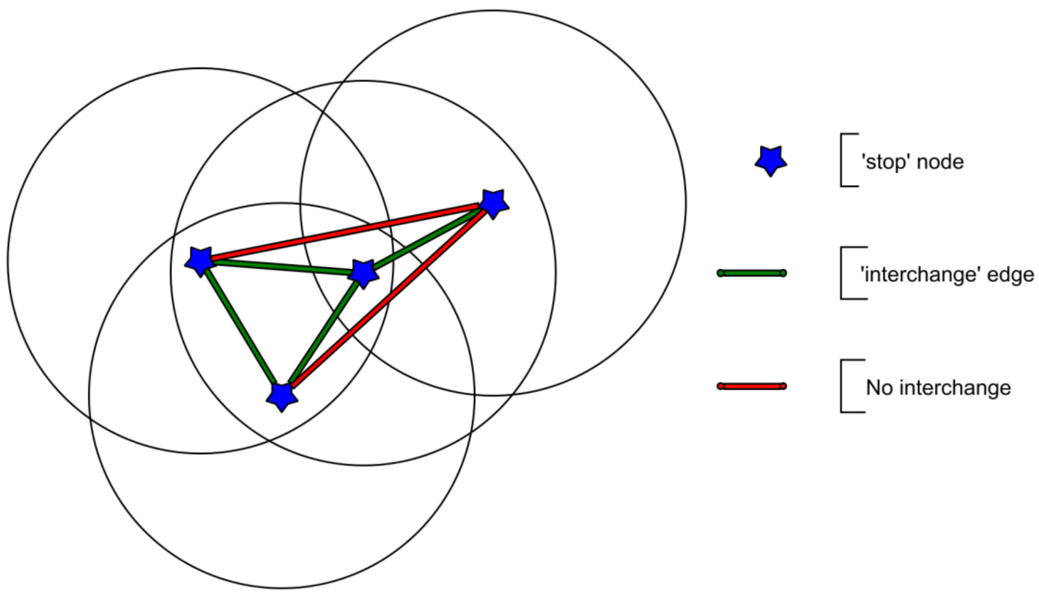

Figure 5.12: Public Interchange Edges

5.7.5 Build Network

NetworkX is used to construct the modified transport network geometrically from a table of nodes containing 'station' nodes and 'centroid' nodes, and a table of edges containing 'route' edges, 'access' edges, and 'interchange' edges.

Processed networks are in the form of a simple graph characterised by a single edge between any given pair of nodes. When networks are constructed from data containing multiple edges between any pair of nodes only the most recently added edge is retained. Since 'interchange' edges are generated regardless of existing 'route' edges between stations it is necessary to upload 'interchange' edges before 'route' edges to ensure that they are overwritten where a connection already exists.

The nx_pgnet library is used to write the processed network to PostgreSQL tables with a specific format supporting graph traversal and pathfinding; network nodes have a unique identifier attribute, whilst network edges have a 'from' node attribute and a 'to' node attribute which link to the node unique identifiers.

5.8 Public Network Cost Model

As shown in Figure 5.13, the public network cost model operates on the processed transport network output from the public network processing model (5.7). The parameters walking weight, waiting weight, average wait, value of time, fare and fare per km are all used in the calculation of generalised cost for each network edge.

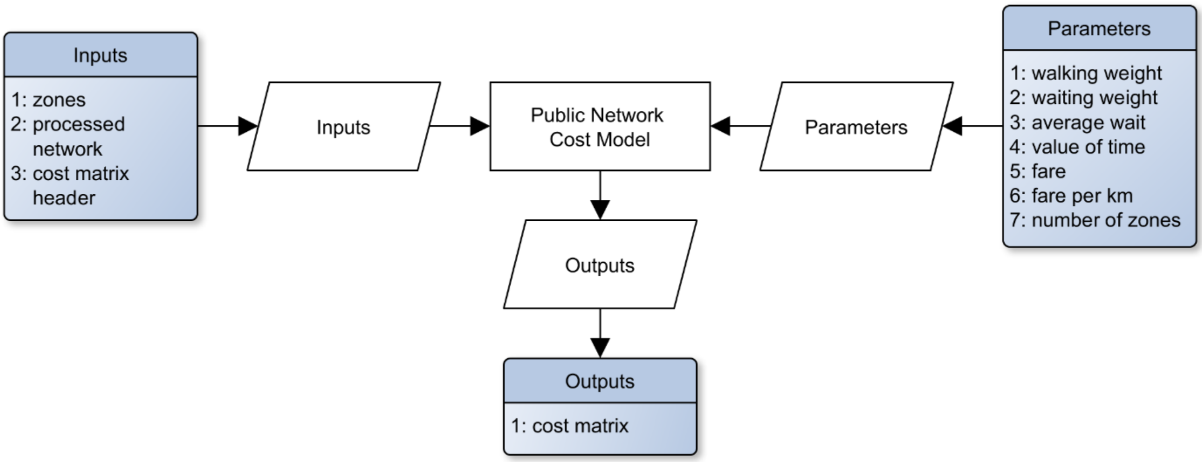

Figure 5.13: Public Network Cost Model Interface

The overall function of the model is to apply input cost parameters to the calculation of weights for each network edge. These weighted edges are then used to compute least cost paths between origin and destination nodes which generates an origin-destination matrix of trip costs. The flow of operations in the model are described in the following subsections as indicated in Figure 5.14.

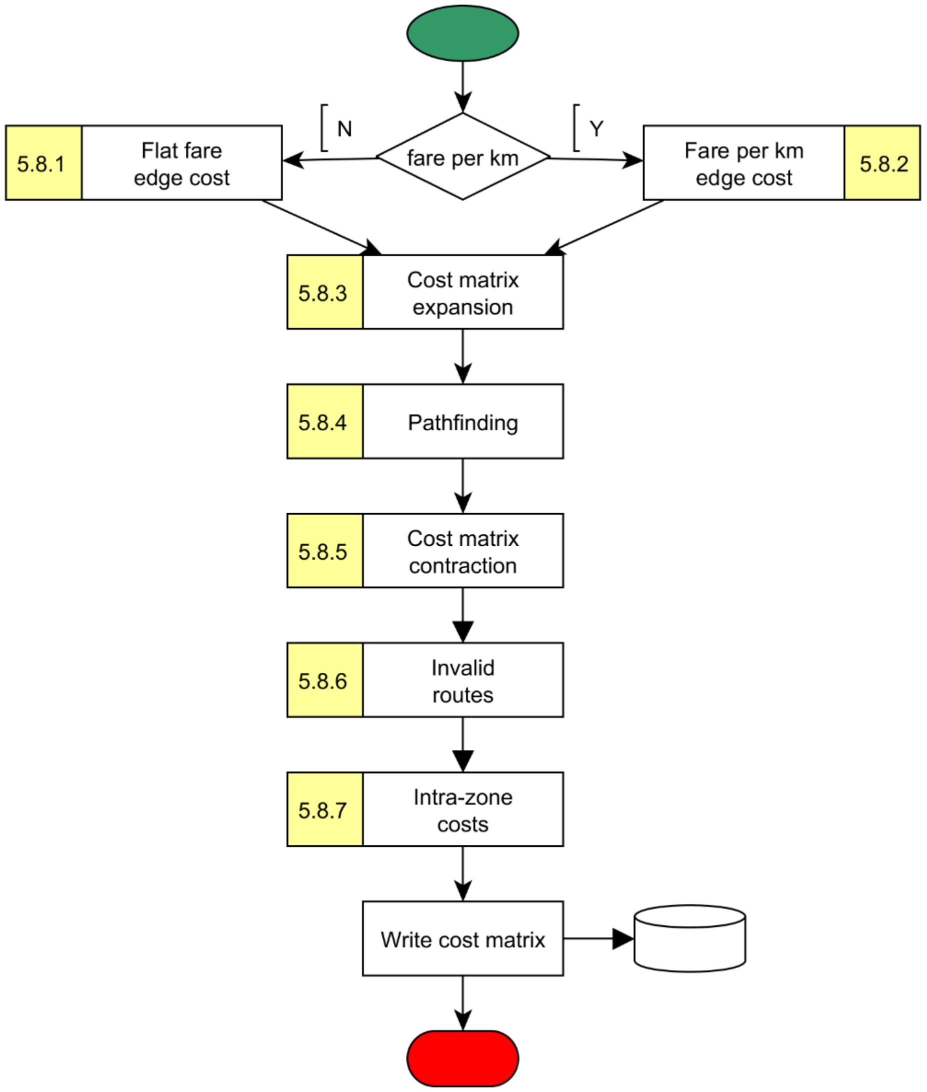

Figure 5.14: Public Network Cost Model Overview

The generalised cost of travel via public transport networks is calculated as shown in Equation 5.14.

$$C_{pub} = (V_{wk} * A) + (V_{wt} * W) + T + \frac{F}{VOT} + I \quad \text{(Eq. 5.14)}$$

Where V_{wk} is the walking disincentive weight; A is the network access time; V_{wt} is the waiting disincentive weight; W is the waiting time; T is the journey time; F is the fare paid; VOT is the value of time; and, I is the interchange penalty.

5.8.1 Flat Fare Edge Cost

The cost of 'access' edges is calculated as shown in Equation 5.15; the cost of 'interchange' edges is calculated as shown in Equation 5.16; and, the cost of 'route' edges is calculated as shown in Equation 5.17.

$$C_{pub}^{access} = V_{wk} * A + V_{wt} * \frac{W}{2} + \frac{F/2}{VOT} \text{ (Eq. 5.15)}$$

$$C_{pub}^{interchange} = V_{wk} * T + V_{wt} * W \text{ (Eq.5.16)}$$

$$C_{pub}^{route} = T \text{ (Eq.5.17)}$$

Where A and T (time) were computed in the public network processing model (5.7) and stored as attributes of network edges.

5.8.2 Fare Per KM Edge Cost

The cost of 'access' edges is calculated as shown in Equation 5.18; the cost of 'interchange' edges is calculated as was shown in Equation 5.16; and, the cost of 'route' edges is calculated as shown in Equation 5.19.

$$C_{pub}^{access} = V_{wk} * A + V_{wt} * \frac{W}{2} \text{ (Eq. 5.18)}$$

$$C_{pub}^{route} = T + \frac{F*D}{VOT} \text{ (Eq.5.19)}$$

Where A and T (time) and D (length) were computed in the public network processing model (5.7) and stored as attributes of network edges.

5.8.3 Cost Matrix Expansion

The order of zones in the origin-destination matrix of trip costs must be consistent with that used by all other models in the framework. A 'Node' data structure with an integer identity and a string label is used to sort 'centroid' nodes by label to enforce zone ordering before copying to a vector of node identities.

The trip cost for zones containing multiple stops is calculated as the average cost of trips via all stops in the zone. In this case, stops rather than centroids are used as trip origins and destinations and must be added to the vector of node identities. Where a zone contains multiple stops, each 'stop' node is added to the vector of node identities and the current index of the vector is recorded to enable retrieval of results when contracting the cost matrix and averaging trip costs. This makes use of the 'from' node and 'to' node attributes of 'access' edges where the 'from' node is the 'centroid' node and the 'to' node is the 'stop' node.

5.8.4 Pathfinding

Network nodes, edges and costs are mapped to arguments of Dijkstra's shortest paths function implemented in the igraph library as shown in Figure 5.15:

1. **Graph construction:** An undirected empty graph with vertices equal to the number of nodes is created before an array of interleaved edge identities is allocated whose size is twice the number of edges. Iterating over all network edges, node identities of 'from' nodes and 'to' nodes are added to the interleaved array which is indexed by an incremental count variable.
2. **Origins and Destinations:** The vector of node identities including ordered 'centroid' nodes and 'stop' nodes (5.8.3) is used.
3. **Edge Weights:** The generalised cost calculated for each edge in the network (5.8.1) (5.8.2) is used as a weight.

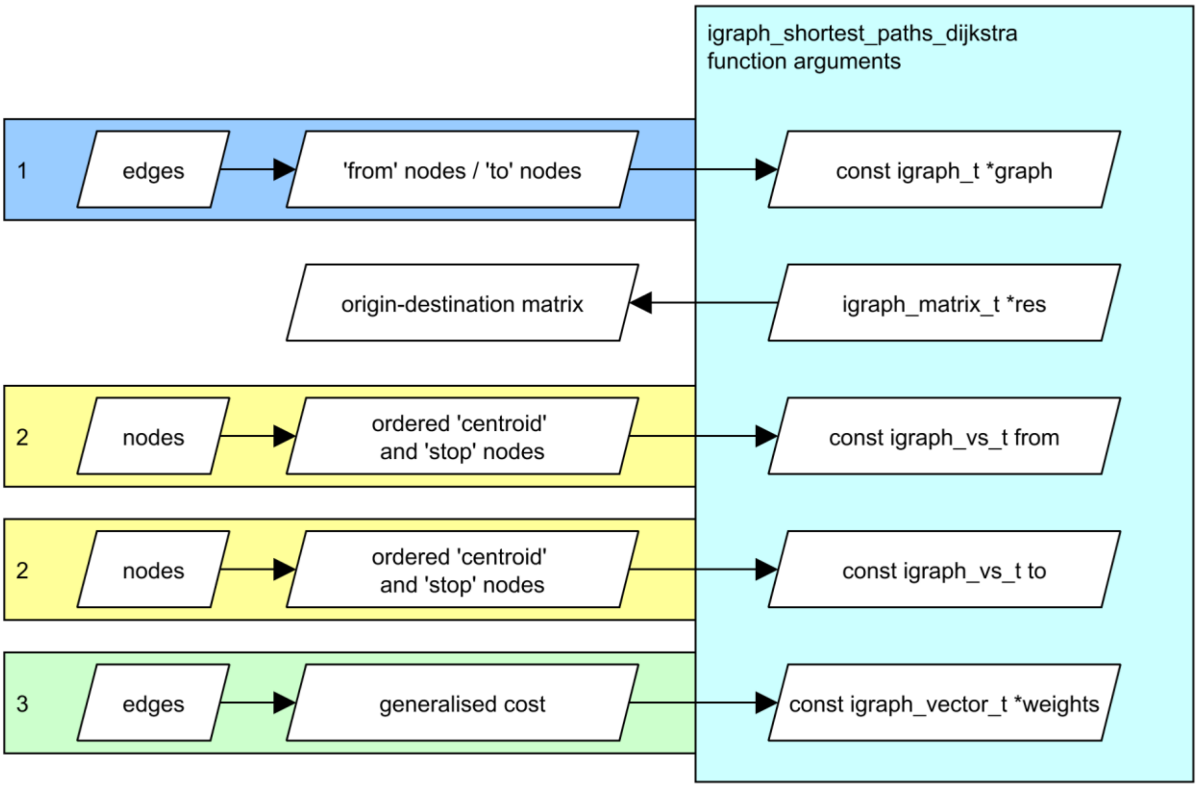

Figure 5.15: Mapping Public Transport Networks to Dijkstra's Algorithm

5.8.5 Cost Matrix Contraction

The trip cost for zones containing multiple stops is calculated as the average cost of trips via all stops in the zone. The expanded cost matrix generated in subsection 5.8.3 holds the results of pathfinding (5.8.4) and must now be contracted to calculate trip costs including averaging where necessary for the output origin-destination cost matrix.

The method of contracting the cost matrix uses a vector of costs to combine and then average trip costs involving multiple routes and is based upon the number of stops and hence ‘access’ edges in each zone as follows:

1. **Neither origin nor destination has multiple stops**
   In this case the expanded cost matrix is copied to the contracted cost matrix at the position of the current origin and destination.
2. **Origin only has multiple stops**
   Iterating over all origin ‘access’ edges, the access cost is read from the ‘access’ edge while the route cost is read from the expanded cost matrix using the ‘access’ edge ‘to node’ as the origin, and the current destination.
3. **Destination only has multiple stops**
   Iterating over all destination ‘access’ edges, the access cost is read from the ‘access’ edge while the route cost is read from the expanded cost matrix using the current origin and the ‘access’ edge ‘to node’ as the destination.
4. **Origin and destination have multiple stops**
   Iterating over all origin ‘access’ edges and all destination ‘access’ edges, the access cost is the sum of the weights read from both origin and destination ‘access’ edges while the route cost is read from the expanded cost matrix using the origin ‘access’ edge ‘to node’ as the origin, and the destination ‘access’ edge ‘to node’ as the destination.

In all multiple stop scenarios, where the route is valid i.e. not the $igraph\_infinity$ value, the route cost is added to the access cost and is pushed into the costs vector. When all ‘access’ edges have been processed, values in the costs vector are summed before dividing by the size of the vector to produce an average cost. This value is then written to the contracted cost matrix at the position of the current origin and destination. An empty costs vector indicates that no valid routes were found in which case the $igraph\_infinity$ value is written to the contracted cost matrix.

5.8.6 Invalid Routes

Removing journeys via shared nearest stations

Iterating over all zones as origins and all zones as destinations, the node identity of the nearest station held within the ‘Edge Group’ for each zone is read at the origin and destination. Where these values are equal the no data value is written to the output cost matrix at the position of the current origin and destination.

Setting no paths

Iterating over all rows and columns of the output cost matrix, cells with the value of $igraph\_infinity$ indicating that no path could be found for a given origin-destination pair are set to the no data value specified in the cost matrix header.

5.8.7 Intra-zone Costs

The intra-zone trip cost (where origin = destination) for any given zone is set at two thirds of the minimum inter-zone trip cost for that zone (Ford et al, 2015) which is found for each origin by searching all destinations in the cost matrix.

5.9 Accessibility Model

As shown in Figure 5.16, the accessibility model operates on input cost matrices for a variable number of transport modes generated by models of Private Network Cost (5.6) and Public Network Cost (5.8). The model outputs an accessibility matrix which combines input cost matrices and converts from cost to accessibility using a deterrence function in a process tuned by parameters lambda λ and beta β. The flow of operations in the model are described in the following subsections as indicated in Figure 5.17.

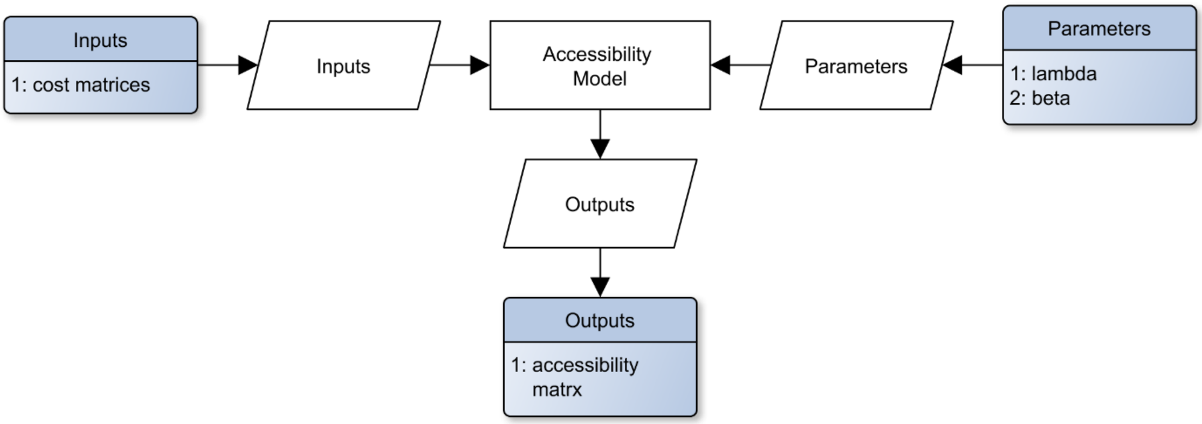

Figure 5.16: Accessibility Model Interface

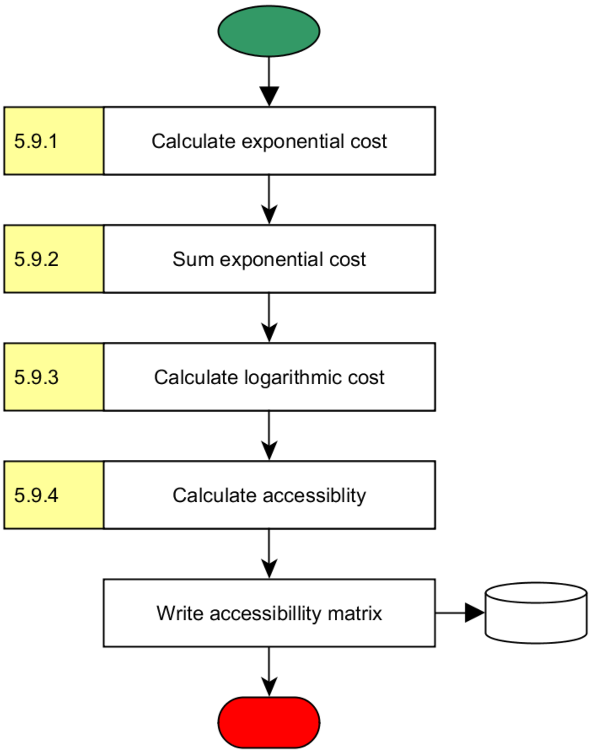

Figure 5.17: Accessibility Model Overview

5.9.1 Calculate Exponential Cost

Iterating over all cells of all input cost matrices, the exponential cost is calculated as shown in equation 5.20. This only applies to cells with positive cost values i.e. where a valid route between origin and destination was found, otherwise the result is set to zero.

$$C_{ij}^{exp} = exp^{-\lambda * C_{ij}} \quad \text{(Eq.5.20)}$$

5.9.2 Sum Exponential Cost

Iterating over all cells of all exponential cost matrices, the current cell of the summed cost matrix is added to by the current cell of each exponential cost matrix as shown in Equation 5.21.

$$C_{ij}^{sum} = \sum C_{ij}^{exp} \quad \text{(Eq.5.21)}$$

5.9.3 Calculate Logarithmic Cost

Iterating over all cells of the summed cost matrix, the current cell is calculated as shown in Equation 5.22. This process is masked to avoid the error associated with finding the logarithm of 0 which is undefined.

$$C_{ij}^{log} = \frac{-1}{\lambda} * \ln\left(C_{ij}^{sum}\right) \quad \text{(Eq.5.22)}$$

5.9.4 Calculate Accessibility

Iterating over all cells of the logarithmic cost matrix, the current cell is calculated as shown in equation 5.23. This process is masked to avoid zero so that only cells where a valid route between origin and destination was found are included, otherwise the result is kept at zero.

$$A_{ij} = e^{-\beta * c_{ij}^{log}} \quad \text{(Eq.5.23)}$$

5.10 Summary

This chapter has described the UIMF TAM which provides a simple method of modelling accessibility using data for transport networks and generalised costs to produce an origin-destination matrix of accessibility. This measure of accessibility is used by the Spatial Interaction Model (SIM – Chapter 4) in combination with zonal employment to form a spatial driver of residential location choice. The TAM allows network modification to simulate the construction of new routes which can be used to explore infrastructure investment in multi-sectoral scenarios using the UIMF in the context of rapid assessment. The UIMF TAM matches the core operation of the UIAF AAF and provides standalone functionality using open source software which is optimised to rapidly generate results. The TAM is integrated within the UIMF which allows the SIM to be driven directly from its outputs to explore patterns of residential location driven by accessibility to employment. These patterns are used in the Urban Development Model (UDM) to model land use at a fine-scale as described in the following chapter.

Chapter 6. Urban Development Model

6.1 Introduction

This chapter presents the Urban Development Model (UDM) implemented in the Urban Integrated Modelling Framework (UIMF). Section 6.2 provides theoretical underpinnings to the simplified approach taken. Section 6.3 describes the development of this model from the reference implementation in the Urban Integrated Assessment Framework (UIAF) and highlights key differences. Section 6.4 gives a high-level overview of the model and its operation within the software environment of the UIMF, and Sections 6.5 to 6.7 provide a detailed description of each stage of the UDM implementation.

6.2 Context

The UDM was developed to downscale predictions of zonal population from the Spatial Interaction Model (SIM) to generate possible fine-scale patterns of land use change resulting from future development. This permits the assessment of these development patterns with respect to environmental models which operate at a finer spatial scale than zone polygons, such as urban drainage and flood risk. The UDM employs spatial Multi-criteria Evaluation (MCE) to determine areas and cells which are most attractive for new development, and a Cellular Automata (CA) spreading function to distribute development across and within development areas which are integrated within existing urban fabric.

6.3 Development

The UIAF UDM was written as a Visual Basic for Applications (VBA) macro using ArcObjects within the ArcGIS environment to manage the loading and manipulation of input spatial data including the conversion from vector to raster datatypes. The UIMF UDM was developed in Python/C++ to match the core UIAF UDM functionality using open source software but operates directly on input raster data using techniques to convert between vector and raster datatypes in QGIS described in Appendix A. These UDM versions are compared in table 6.1.

Feature	UIAF UDM	UIMF UDM
Model performance	Unknown	Optimised using C/C++
Link to spatial interaction model	Manual loose coupling	Integrated coupling via UIMF model group chaining
Metadata	No	Yes
Graphical User Interface	Yes	No
Open source	No	Yes
Vector to raster conversion	Internal	External

Table 6.1: UDM Version Comparison

6.4 Overview

The urban development model (figure 6.1) disaggregates zonal population to cellular urban development which is integrated with existing land-use. It comprises 3 stages whose raster inputs and outputs are internally linked as shown in figure 6.2. The first stage (6.5) employs multi criteria evaluation to produce a raster of development suitability from weighted suitability inputs and development constraints. The second stage (6.6) groups cells into development areas within each zone of a specified minimum size and computes the average development suitability for each area using the raster output from multi criteria evaluation. The final stage (6.7) develops cells within development areas in each zone as required by zonal population change. This process makes use of suitability for both development areas, and individual cells to distribute development and produce a raster output showing current and future development, as well as undeveloped cells.

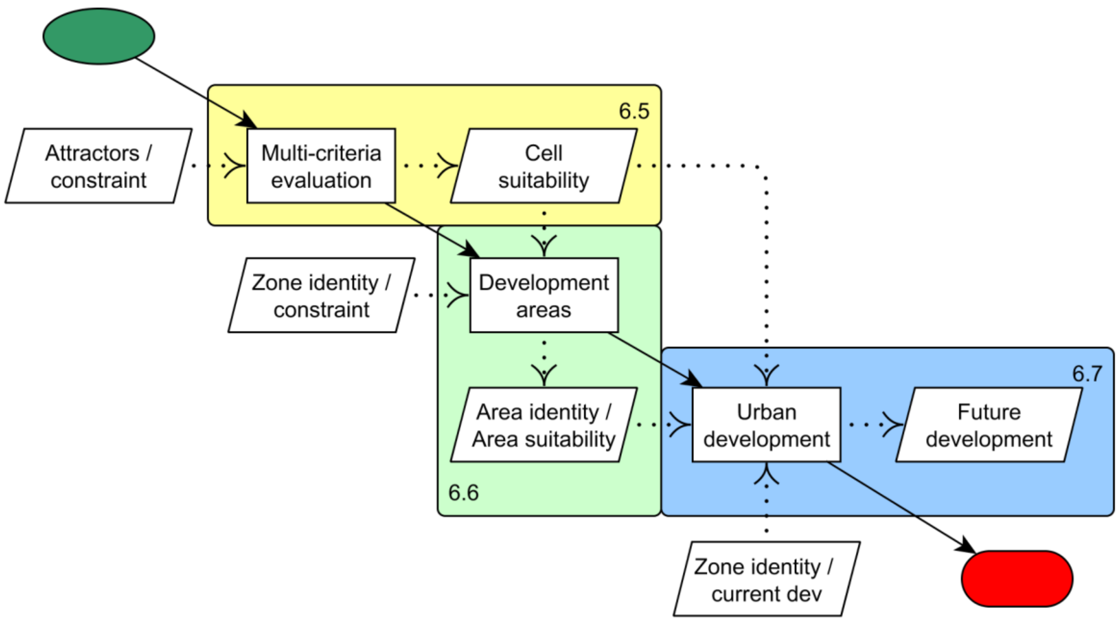

Figure 6.1: Urban Development Model Overview

The inputs, outputs and parameters shown in figure 6.2 are used by the model stages as follows:

- All stages of the UDM use the input ***'raster header'*** to setup, load and save raster data.
- The multi-criteria evaluation stage takes input tables specifying input raster data of integer and double precision type; ***'MCE rasters'***. Integer raster data is used for the constraint mask and discrete attractors, whilst double precision raster data is used for continuous attractors such as rasterised proximity. Each raster input has an associated weight which is used to produce a masked weighted summation of development suitability.
- The creation of development areas within zones is controlled by the ***'minimum dev area'*** parameter which is specified in terms of the number of cells. This parameter is used in conjunction with cell size derived from the raster header to configure a suitable minimum development area size. A further parameter, ***'expanded neighbourhood'*** dictates the type of cell neighbourhood used when grouping cells into development areas: a value of true specifies an 8-cell Moore neighbourhood, whilst false specifies a 4-cell Von Neumann neighbourhood. Raster inputs are required in the form of the constraint mask and zone identity to create development areas within zones; this raster data is specified in the input table ***'UDM rasters'***.
- In addition to the raster data generated by the previous stages, the final urban development stage takes input raster data in the form of zone identity and current development, both of which are referenced in the input table ***'UDM rasters'***. The input ***'zones'*** is used to provide labels for overflow data while the input ***'zonal population'*** holds a population value for each zone in each time-step. This data is used to calculate the change in population and hence required development for each zone. The input table ***'density'*** is used in conjunction with the parameter ***'density provided'*** to provide density values to the urban development model which are consistent with the zonal model.

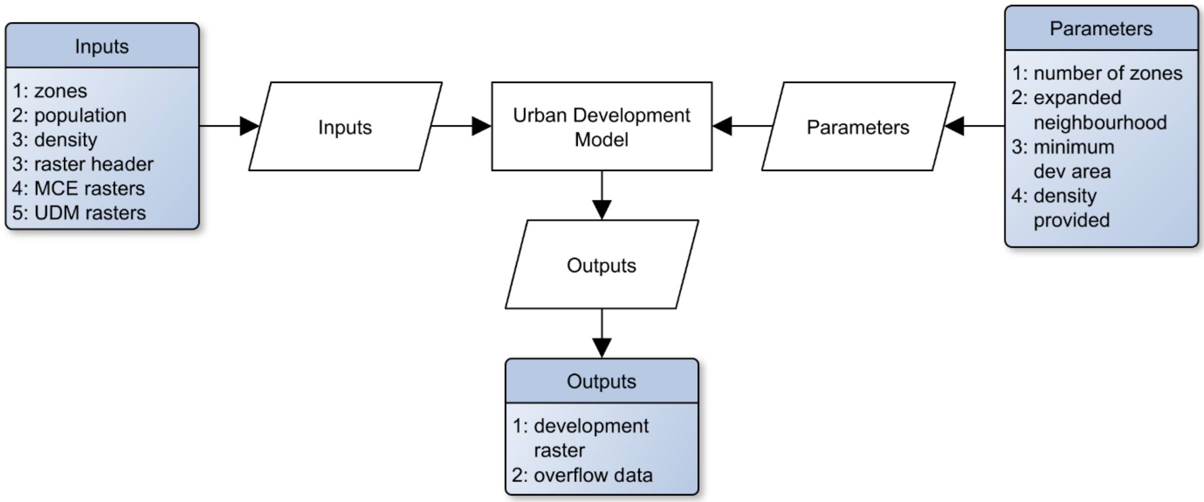

Figure 6.2: Urban Development Model Interface

6.5 Multi-Criteria Evaluation

The multi criteria evaluation model operates on a variable number of raster inputs named in tables also containing weights data for each input raster. Raster tables have associated row count tables from which the number of rasters and weights is obtained and used for dynamic memory allocation.

The MCE model performs the following operations:

1. **Output raster setup**

   A result raster is setup using the input raster header and all cells are initialised to zero.

2. **Input raster reading**

   The MCE dynamically allocates arrays for raster names and weights which are read from the input raster tables. A vector of rasters is setup and the specified number of (empty) rasters are pushed in. This vector is then iterated over, setting up each raster and loading cell data from files specified in the raster names array.

3. **Weighted summation**

   A weight value of -1 is used to identify the constraint mask which is excluded from the following summation: all cells in each raster are iterated over and the value read from each input raster is multiplied by the corresponding raster weight before being added to the result raster.

4 Masking

Iterating over all cells, if the constraint raster is invalid at the current cell position i.e. has the no data value, this is reproduced in the result raster; otherwise, the result raster is multiplied by the constraint raster (0 or 1) to mask the weighted summation.

5 Raster output

The final masked weighted sum raster is then written to file for use in further stages (6.6 and 6.7) of the UDM.

6.6 Development Areas

Figure 6.3 shows the flow of operations to create development areas which makes use of two simple data structures to group cells into areas. The first defines a 'Cell' which holds variables for row, column, and area identity. The second defines an 'Area' which holds an identity variable, and a vector of Cells.

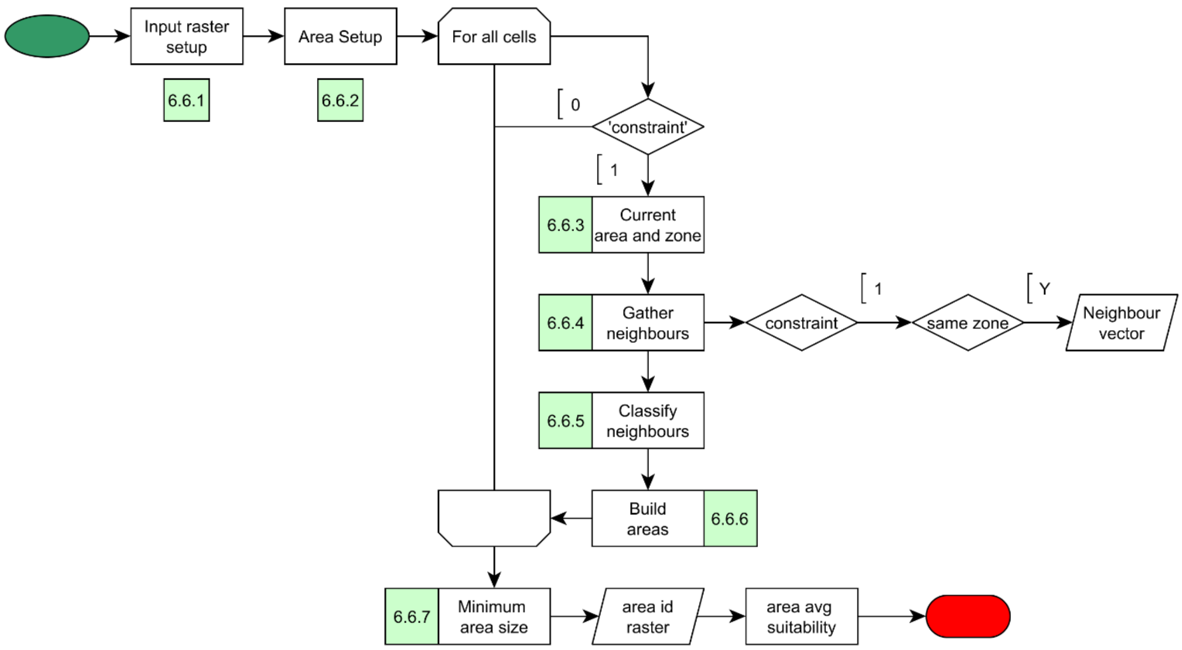

Figure 6.3: Development Area Creation

6.6.1 Input raster setup

There are two raster inputs to the algorithm in the form of zone identity, and a constraint mask as used in multi criteria evaluation. Areas are created by grouping together contiguous unconstrained cells which are in the same zone. The constraint mask is declared and setup using the input raster header before loading cell data from file via the input raster string.

If the creation of development areas does not consider zone boundaries i.e. the spatial extent of a given area is not constrained to a single zone, then errors can occur when spreading urban development so a zone identity raster is taken as an input which is loaded in the same manner as the constraint mask.

6.6.2 Area Setup

Initial and final area identity rasters are setup using the input raster header and their cells are initialised to -1 which is defined as 'NOAREA'. The initial raster is used to keep track of areas as they are created whilst the final raster is used to output results after the minimum area size parameter has been applied. An area identity count variable is set to 0 to identify the first area and is then incremented after each new area is created. A vector of type 'Area', $V_{(area)}^{complete}$, is declared which will hold all cells in all areas when the algorithm is complete.

6.6.3 Current area and zone

Raster cells are indexed by row and column, and each cell is designated as the current cell exactly once starting at the top left of the defined raster region. Nested loops are used to iterate over all cells in the raster, for which row and column values are used to access and store data. Constrained cells are excluded from development areas and the first use of the row and column iteration values is to index the constraint raster at the current position. A mask value of 1 indicates unconstrained, and processing of the current cell continues whilst a value of 0 indicates constrained, and the algorithm skips to the next cell iteration. Further values are required to define the current position in the form of the current area which is read from the initial area identity raster, and the current zone which is read from the zone identity raster. The current area value is used along with the row/column iteration values to form an instance of the simple 'Cell' data structure.

6.6.4 Gather Neighbours

The algorithm then proceeds to examine cells contiguous to the current position. The expanded neighbourhood parameter is used to specify the contiguity type where false indicates a 4-cell Von Neumann neighbourhood, and true indicates an 8-cell Moore neighbourhood as depicted in figure 6.4. These neighbouring cells are accessed by offsetting the current row and/or column iteration values by plus or minus one provided the offset value is a valid raster cell index. A vector of type 'Cell', $V_{(cell)}^{nbrs}$, is used to store neighbouring cells which are both unconstrained (read from constraint mask) and in the same zone as the current cell (read from zone identity and compared with current zone). The area value of cells meeting these criteria is read from the initial area identity raster before forming a 'Cell' instance and pushing into the neighbourhood cells vector, $V_{(cell)}^{nbrs}$.

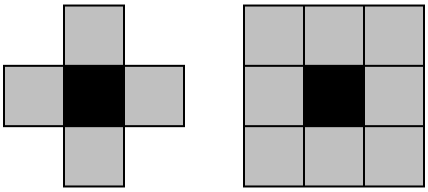

Figure 6.4: Von Neumann (left) and Moore (right) cell neighbourhoods

6.6.5 Classify Neighbours

This algorithm builds all areas incrementally, adding cells and merging partially constructed areas via the current cell position as it moves from top left to bottom right. It is necessary therefore to consider the function of the current cell as either seeding a new area identity and passing this to a neighbour, inheriting an area identity from a neighbour, or bridging between two sections of the same area which currently have different area identities.

The neighbourhood cells vector, $V_{(cell)}^{nbrs}$, contains a maximum of 8 cells which are first classified by their area status as depicted in figure 6.5. Two further 'Cell' vectors; area cells, $V_{(cell)}^{area}$, and no area cells, $V_{(cell)}^{no\ area}$; are used to divide the neighbourhood cells vector, $V_{(cell)}^{nbrs}$, into cells with either a valid area identity or the 'NOAREA' initialised identity. This distinction allows the current cell to either seed or inherit an area identity as appropriate. The area cells vector holds cells which may have different area identities. To merge areas via the current cell the lowest area identity observed is assigned to the current cell and to all cells contained within areas to be merged i.e. with a greater area identity. Two further 'Cell' vectors; low area cells, $V_{(cell)}^{low}$, and high area cells, $V_{(cell)}^{high}$; are used to divide the area cells vector, $V_{(cell)}^{area}$, by area identity. The lowest observed identity is recorded and used to split the area cells vector, $V_{(cell)}^{area}$, where cells with area identities higher than the recorded value are pushed into the high area cells vector, $V_{(cell)}^{high}$, or otherwise the low area cells vector, $V_{(cell)}^{low}$. Finally, a vector of integer merge area identities, $V_{(id)}^{merge}$, is created and populated with unique area identities from the high area cells vector, $V_{(cell)}^{high}$, and from the current cell where this has already been assigned a higher area identity.

6.6.6 Build Areas

The algorithm builds and merges areas based on the size of the various 'Cell' vectors holding neighbouring cells. When a cell seeds an area, it is assigned the area identity count value which is also written to the initial area identity raster at the cell position. A new area is created using this value and pushed into the areas vector before the cell is pushed into the area. When a cell joins an area, it is assigned the area identity of a cell currently in the area which is also written to the initial area identity raster at the cell position. The cell is then pushed into the area. When a cell moves area, it joins its new area and is removed from its previous area.

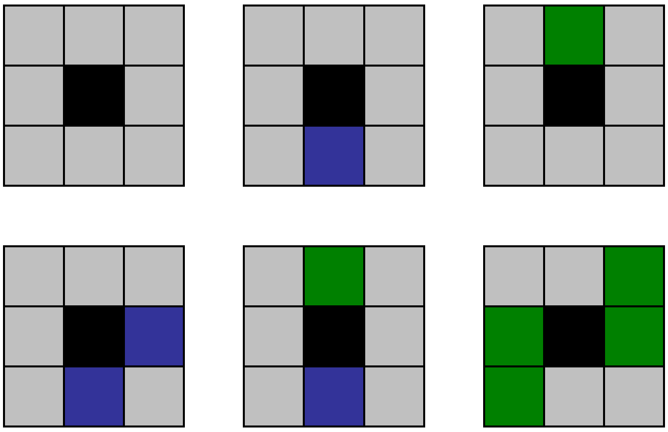

Figure 6.5: Neighbouring cell classification

No neighbours (figure 6.5, **top left)**

The neighbourhood cells vector, $V_{(cell)}^{nbrs}$, is empty and the current cell seeds a new area before incrementing the area identity count value. A new area is created so that the algorithm can be used with a minimum area size of a single cell.

Single neighbour, not currently in an area (figure 6.5**, top middle)**

The neighbourhood cells vector, $V_{(cell)}^{nbrs}$, contains a single cell and the area cells vector, $V_{(cell)}^{area}$, is empty. The current cell seeds a new area and the single cell in the no area cells vector, $V_{(cell)}^{no\ area}$, joins the area before incrementing the area identity count value.

Single neighbour, currently in an area (figure 6.5**, top right)**

The neighbourhood cells vector, $V_{(cell)}^{nbrs}$, contains a single cell and the no area cells vector, $V_{(cell)}^{no\ area}$, is empty. The area identity of the current cell is tested against the 'NOAREA' value to find out if it is already in an area. If the current cell is not in an area, it joins the area of the single cell in the area cells vector, $V_{(cell)}^{area}$.

Multiple neighbours, none currently in an area (figure 6.5**, bottom left)**

The neighbourhood cells vector, $V_{(cell)}^{nbrs}$, contains multiple cells and the area cells vector, $V_{(cell)}^{area}$, is empty. The current cell seeds a new area and all cells in the no area cells vector, $V_{(cell)}^{no\ area}$, join the area before incrementing the area identity count value.

Multiple neighbours, one currently in an area (figure 6.5**, bottom middle)**

The neighbourhood cells vector, $V_{(cell)}^{nbrs}$, contains multiple cells and the area cells vector, $V_{(cell)}^{area}$, contains a single cell. The area identity of the current cell is tested against the 'NOAREA' value to find out if it is already in an area. If the current cell is not in an area, it joins the area of the single cell in the area cells vector, $V_{(cell)}^{area}$. All cells in the no area cells vector, $V_{(cell)}^{no\ area}$, join the area of the single cell in the area cells vector, $V_{(cell)}^{area}$.

Multiple neighbours, multiple currently in an area (figure 6.5**, bottom right)**

The neighbourhood cells vector, $V_{(cell)}^{nbrs}$, contains multiple cells and the area cells vector, $V_{(cell)}^{area}$, contains multiple cells. The area identity of the current cell is tested against the 'NOAREA' value to find out if it is already in an area. If the current cell is not in an area, it joins the area of the first cell in the low area cells vector, $V_{(cell)}^{low}$. All cells in the no area cells vector, $V_{(cell)}^{no\ area}$, join the area of the first cell in the low area cells vector, $V_{(cell)}^{low}$. If the high area cells vector, $V_{(cell)}^{high}$, is not empty, the merge area ids vector, $V_{(id)}^{merge}$, is also not empty, and all cells in all areas indexed in the merge area identities vector, $V_{(id)}^{merge}$, move to the area of the first cell in the low area cells vector, $V_{(cell)}^{low}$.

6.6.7 Minimum Area Size

The iteration over all cells is now complete so the vector, $V_{(area)}^{complete}$, contains all unconstrained cells in all areas. The minimum area size parameter can now be applied whilst tidying up area identity values using an output area identity count which is initially set to zero. Iterating over all areas, if the size of the area's cell vector is greater than or equal to the minimum area size then the current output area identity count value is written to the final area identity raster at the position of each cell before being incremented between valid areas. The final area identity raster is then written to file and is used to compute average development area suitability along with the output from the MCE stage (6.5).

6.6.8 Development Area Suitability

Figure 6.6 shows the flow of operations to calculate the average suitability of development areas which makes use of two simple data structures to group cells into areas. The first defines a 'Cell' which holds variables for row, column, and area. The second defines an 'Area' which holds an identity variable, and a vector of Cells. The variable 'NOAREA' is defined as -1.

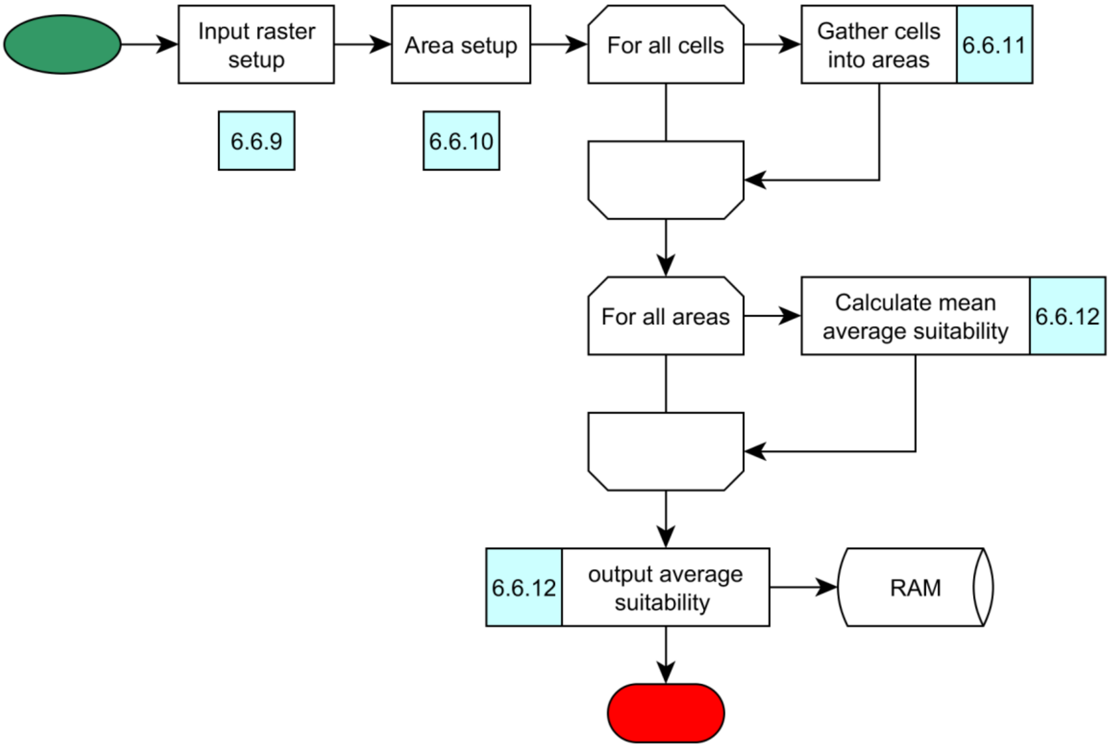

Figure 6.6: Development Area Average Suitability

6.6.9 Input raster setup

There are two raster inputs to the algorithm in the form of area identity (output by development area creation), and development suitability (output by multi criteria evaluation). Both rasters are declared and setup using the input raster header before loading cell data from file via the input raster strings.

6.6.10 Area setup

An area suitability raster is setup using the input raster header and its cells are initialised to zero. An area count variable is set to zero, which will be incremented after each new area is found. A vector of type 'Area', $V_{(area)}^{found}$, is declared which will hold all cells in all areas when the algorithm has read in all areas from the area identity raster.

6.6.11 Gather Cells into Areas

Raster cells are indexed by row and column values starting at zero. To set the limits for cell iteration, the total number of rows and columns are obtained from the area identity raster. Nested loops are used to iterate over all cells in the raster, for which row and column values are used to access and store data. The method used for area creation results in an area identity raster in which the identity of cells in new areas increments sequentially from zero when reading data from the top left of the raster, in a rightwards and downwards manner. This makes it straightforward to assign cells to areas which is the first stage of this algorithm. Iterating over all cells, the area identity is read from the area identity raster at the current cell position. Where area identity is greater than 'NOAREA', the cell belongs to a valid area. Where area identity is equal to the size of the found areas vector, $V_{(area)}^{found}$, a new area is created with the area identity and pushed into the found areas vector, $V_{(area)}^{found}$, before incrementing the area count. Cells are then pushed into the area with corresponding area identity.

6.6.12 Mean average area suitability

The found areas vector, $V_{(area)}^{found}$, now contains all cells in all areas, and the area count records the total number of areas found. The area count value is used to dynamically allocate an array of double precision variables to hold area average values; this is populated for each area by summing the suitability values for all cells (read from the development suitability raster) and dividing by the number of cells in the area.

6.6.13 Output Average Suitability

The area average values are then written to the area suitability raster by iterating over all cells. Area identity is read from the area identity raster at the current cell position. Where area identity is greater than 'NOAREA', the cell belongs to a valid area and the area average value (indexed by area identity) is written to the area suitability raster at the current cell position. Where area identity is 'NOAREA' and the current value read from the development suitability raster is also 'NOAREA' then 'NOAREA' is written to the area suitability raster. The final development area suitability raster is then written to file for use in the final stage of the UDM (6.7).

6.7 Urban Development

Figure 6.7 shows the flow of operations in the final stage of the UDM which makes use of a hierarchical structure of classes to manage the development process. At the top level of this class hierarchy is a vector of type ‘Zone’, $V_{(zone)}$,. The ‘Zone’ class houses zone-level variables, such as population change, as well as a vector of type ‘Area’, $V_{(area)}$, and a vector of type ‘Cell’, $V_{(cell)}$. The ‘Area’ class has its own vector of type ‘Cell’, $V_{(cell)}$.

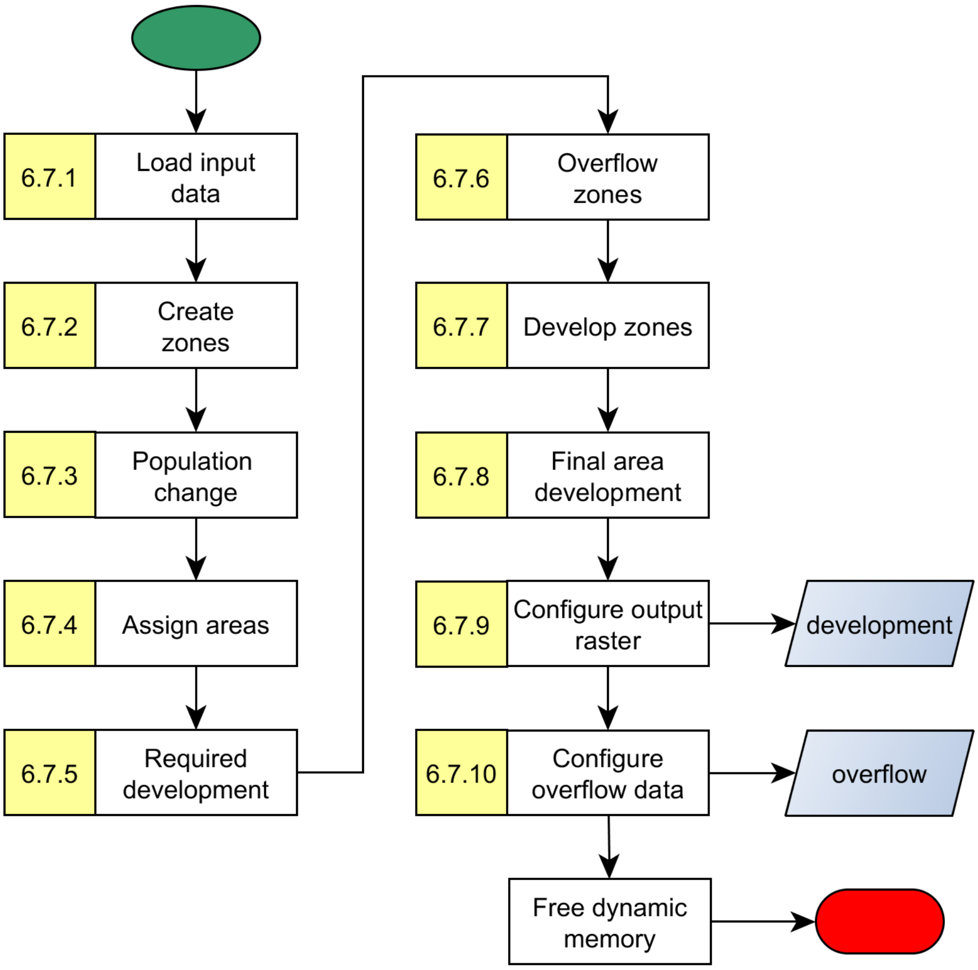

Figure 6.7: Urban Development

6.7.1 Load Input Data

Urban development requires 5 input rasters containing the following cell data: zone identity, current development, area identity (from development area creation), area suitability (from area average suitability), and cell suitability (from multi-criteria evaluation). All rasters are declared and setup using the input raster header before loading cell data from file via the input raster strings.

6.7.2 Create Zones

The number of zones parameter is used to determine the number of empty zones which are pushed into the zones vector, $V_{(zone)}$. Iterating over all cells using the number of rows and columns, the zone identity raster is read at the current cell position. Where this is valid (i.e. not the no data value) the value returned is used as an index to the zones vector, $V_{(zone)}$, and the current cell is pushed into the vector of cells, $V_{(cell)}$, within that zone, $V_{(zone)}$.

6.7.3 Population change

Iterating over all zones, $V_{(zone)}$, the zone's population change is calculated by subtracting current population from future population using the iteration number to index the double precision arrays loaded earlier. Where population change is less than or equal to zero the zone's Boolean variable, development required, is set to false.

6.7.4 Assign areas

Iterating over the vector of cells, $V_{(cell)}$, in all zones, $V_{(zone)}$, which require development, the area identity at the current cell position is read from the area identity raster. Where area identity is valid (i.e. not the no data value) the vector of areas, $V_{(area)}$, in the zone, $V_{(zone)}$, is searched for the area identity value. Where an area with corresponding area identity is found, the cell is pushed into the vector of cells, $V_{(cell)}$, within that area, $V_{(area)}$, otherwise the cell is pushed into a new area; areas are created using the area identity value and the area suitability value which is read from the area suitability raster at the current cell position.

6.7.5 Required development

Iterating over the vector of cells, $V_{(cell)}$, in all zones, $V_{(zone)}$, which require development, the current development raster is read at the current cell position. A value of 1 indicates a currently developed cell which is added to a count of developed cells for each zone, $V_{(zone)}$, and used to set the zone's current development cells variable, $Z^{cur\ dev\ cells}$.

Zonal current development area, $Z^{cur\ dev\ area}$, is found by multiplying the current development cells, $Z^{cur\ dev\ cells}$, by the cell size squared.

Where density values are provided for each zone:

- Zonal density, $Z^{density}$, is read from the array of provided data.
- Zonal cell density, $Z^{cell\ density}$, is found by multiplying the density, $Z^{density}$, by the cell size squared.

Otherwise:

- Zonal cell density, $Z^{cell\ density}$, is found by dividing the current zone population, $Z^{cur\ pop}$, by the current development cells, $Z^{cur\ dev\ cells}$.
- Zonal density, $Z^{density}$, is found by dividing the current zone population, $Z^{cur\ pop}$, by the current development area, $Z^{cur\ dev\ area}$.

Future development is found by dividing the zonal population change, $Z^{pop\ change}$, by the zonal cell density, $Z^{cell\ density}$. This value is then rounded up to the nearest integer and set as the zonal required development cells, $Z^{req\ dev\ cells}$.

6.7.6 Overflow zones

Iterating over all zones, $V_{(zone)}$, which require development, the size of the vector of cells, $V_{(cell)}$, in all areas, $V_{(area)}$, within the zone, $V_{(zone)}$, are summed to produce the zone's suitable development cells variable. If the zone's suitable development cells variable is less than the zone's required development variable the zone's Boolean overflow variable is set to true.

6.7.7 Develop zones

Overflow zones are developed by iterating over all overflow zones, $V_{(zone)}$, which require development, the development status of all cells, $V_{(cell)}$, in all areas, $V_{(area)}$, is set to true, incrementing the zone's developed cells variable in the process.

Iterating over all non-overflow zones, $V_{(zone)}$, which require development, areas, $V_{(area)}$, within zones, $V_{(zone)}$, are sorted by highest average area suitability as depicted in figure 6.8. Iterating over all areas, $V_{(area)}$, within each zone, $V_{(zone)}$, which requires further development (where the zone's developed cells variable is less than the zone's required development cells variable); the algorithm establishes if the current area, $V_{(area)}$, is the final area where development spreading takes place by checking whether the zone's developed cells variable plus the size of the cell vector, $V_{(cell)}$, in the current area, $V_{(area)}$, is greater than the zone's required development cells variable. Where this is false the current area, $V_{(area)}$, is not the final area and all cells within the area are developed by iterating over the cells vector, $V_{(cell)}$, setting each cell's Boolean development status variable to true, and incrementing the zone's developed cells variable.

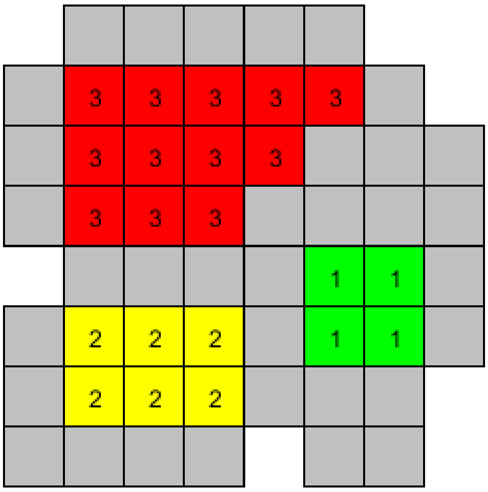

Figure 6.8: Areas ranked and developed by average suitability

6.7.8 Final area development

Carrying on from the previous section, the algorithm is in an iterative cycle of all non-overflow zones, $V_{(zone)}$, which require development and the current area, $V_{(area)}$, is the final area where development spreading takes place. Iterating over all cells, $V_{(cell)}$, within the final spreading area, $V_{(area)}$, the cell suitability is read from the cell suitability raster at the current cell position. Cells within the final area are then sorted by highest suitability placing the most suitable, the seed cell, at the front of the cells vector, $V_{(cell)}$. Two further vectors of type 'Cell' are setup to store cells used as seeds, $V_{(cell)}^{seeds}$, and neighbours, $V_{(cell)}^{nbrs}$, the initial seed cell is developed and the zone's developed cells variable is incremented before the initial seed cell is pushed into the vector of seed cells, $V_{(cell)}^{seeds}$.

Since it is known in advance that there are sufficient cells to meet the zone's required development cells variable, a while loop is used to control the spreading algorithm depicted in figure 6.9, while the zone's developed cells variable is less than the zone's required development cells variable.

Within this loop, the neighbours cell vector, $V_{(cell)}^{nbrs}$, is cleared and then populated with cells within the area which make up an 8-cell Von Neumann neighbourhood around the final cell in the seeds vector, $V_{(cell)}^{seeds}$, then the current seed is removed by popping the back of the seeds vector, $V_{(cell)}^{seeds}$. Cells in the neighbourhood vector, $V_{(cell)}^{nbrs}$, are sorted by highest suitability and developed in turn where the cell has not already been developed and the zone's required development cells variable has not been reached, whilst incrementing the zone's developed cells variable. Cells in the neighbourhood vector, $V^{nbrs}_{(cell)}$, are then pushed into the seeds vector, $V^{seeds}_{(cell)}$, which is sorted by lowest suitability so that the most suitable seed can be accessed via the final position in the vector and can then be removed by popping the back of the seeds vector, $V^{seeds}_{(cell)}$.

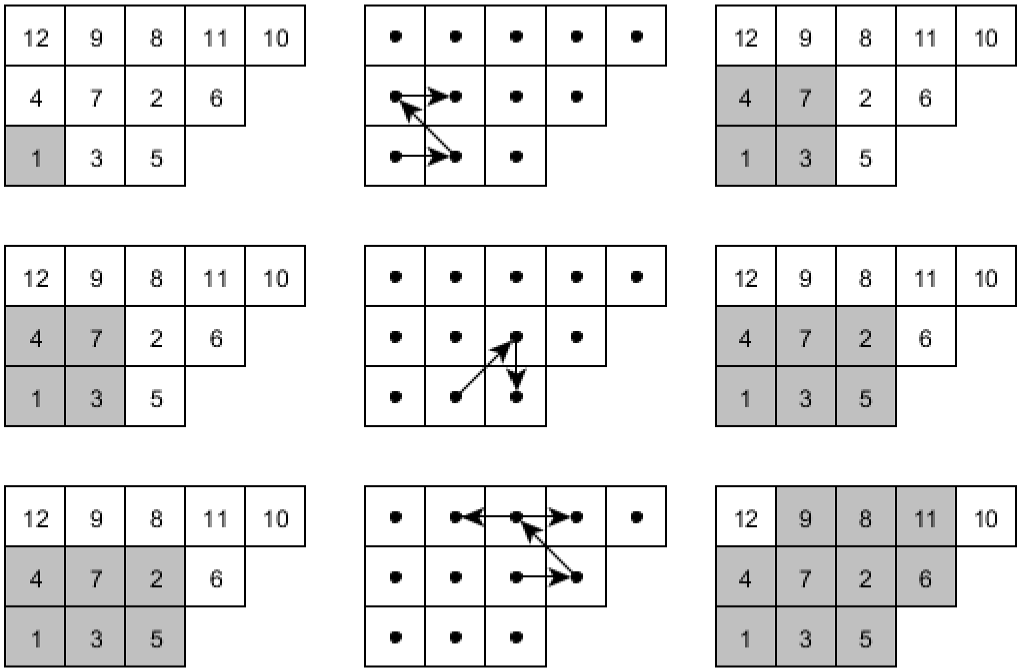

Figure 6.9: Final Area Seeding and Spreading

6.7.9 Configure output raster

The raster output from the UDM has a single band with integer values signifying land use status as follows:

- No Development.

Iterating over all cells, if the value read from the zone identity raster is the no data value then the value written to the final development raster is the no data value, otherwise it is initialised to 0.

- Current Development.

Iterating over all cells, the final development raster at the current cell position is set to the corresponding value read from the current development raster where this is equal to 1 signifying current development.

- Future Development.

Iterating over all cells, $V_{(cell)}$, in all areas, $V_{(area)}$, in all zones, $V_{(zone)}$, which require development, if the cell development status is true then a value of 2 is written to the final development raster at the current cell position indicating future development.

6.7.10 Configure overflow data

The following attributes are output for each zone:

- Label
- Overflow status
- Current population
- Future population
- Population assigned
- Population not assigned
- Expected number of cells developed
- Actual number of cells developed
- Area required
- Area developed
- Current cell density
- Required density in free cells
- Required density in zone cells

6.8 Summary

This chapter has described the UIMF UDM which provides a simple method to downscale zonal population output from the Spatial Interaction Model (SIM) generating fine-scale patterns of land use change resulting from future development. The model permits the assessment of land use patterns in conjunction with environmental models which operate at a finer spatial scale than zone polygons. The UDM employs spatial Multi-criteria Evaluation (MCE) to determine areas and cells which are most attractive for new development, and a Cellular Automata (CA) spreading function to distribute development across and within development areas which are integrated within existing urban fabric. The UIMF UDM matches the core operation of the UIAF UDM and provides standalone functionality using open source software which is optimised to rapidly generate results and is integrated within the UIMF which allows it to be driven directly from outputs of the SIM.

The following chapter presents a baseline case study applied to the Greater London area which demonstrates the application of the UIMF using a rich set of parameterisation data representing policies set out in the London Plan.

Chapter 7. London Case Study

7.1 Introduction

This chapter revisits the original setting of the Urban Integrated Assessment Framework (UIAF) models as developed by the Tyndall Centre for Climate Change Research. The case study is focussed on the Greater London Authority (GLA) area with a population of 8.2 million people at the time of the 2011 Census (GLA, 2016). The activity and transport demand of this densely populated region generate huge greenhouse gas (GHG) emissions, 42.5 million tonnes of carbon dioxide in 2009 (LEGGI), which must be reduced by mitigation strategies. The GLA area is situated in the southeast of the UK, a region vulnerable to heat waves, water scarcity and sea level rise. Potential climate change impacts within the GLA area such as flooding, water shortages, excessive urban heat, and air quality problems require adaptation strategies to minimise risk to the urban population.

The purpose of this case study is to demonstrate that the Urban Integrated Modelling Framework (UIMF) can be parameterised for a baseline scenario in the GLA area using existing transport networks and detailed spatial development strategy as described in section 7.2. This shows the utility of the framework when working with generic model groups, interfaces and datasets, providing evidence that the recoded models have been implemented successfully by demonstrating their combination and the results produced for the GLA area when running under the framework.

Section 7.2 introduces the London Plan and the key spatial development policies which are to be used as model drivers; sections 7.3 to 7.5 cover model parameterisation, section 7.6 describes the process of model and model group setup within the framework, finally, section 7.7 presents the results generated by each modelling stage from transportation to zonal and sub-zonal land use disaggregation.

7.2 Planning Policy

The GLA was formed in 2000 and has adopted a proactive approach to dealing with climate change mitigation and adaptation at the urban level, including collaborating with the Tyndall Centre to develop questions requiring new models to provide insights. The London Plan published by the GLA disseminates the social, economic, environmental and spatial development framework for the region; the plan is a statutory requirement, first published in 2004, and most recently in 2017 as an amendment to the current plan of 2016. In all published versions to date, chapter two describes the broad development strategy and culminates by drawing together spatial policies in a key diagram as shown in figure 7.1.

The 8.2 million people living in the GLA area at the time of the 2011 Census were housed in 3.28 million households and population was growing at an average rate of 87 thousand people per annum relative to the 2001 Census. Although subject to a degree of uncertainty, in some cases the population is projected to rise to 9.2 million people in 3.74 million households by 2021 and reach 10.1 million people in 4.26 million households by 2036. Population growth on this scale requires detailed planning to provide housing, jobs and infrastructure to an expanded urban population whose needs must be met without negatively impacting upon quality of life. Growth in the economy is expected to be accompanied by change as new sectors develop whose specific needs in terms of locations and infrastructure should be fostered. Climate change adaptation, including the protection of urban green space and management of flood risks must be progressed alongside developments to reduce GHG emissions as required by mitigation (GLA, 2016).

The spatial policies set out in the Key Diagram of the London Plan (figure 7.1) form inputs to the multi-scale simulation framework as described in the following sub-sections.

Opportunity areas have the potential to accommodate a sizable increase in employment (~5000 jobs) or housing (~2500 homes). Most include large areas of previously developed land and offer significant potential for densification. They hold promise for climate change adaptation and mitigation measures with their development supported by improvements to public transport access where required. **Areas of intensification** have good public transport accessibility and offer the potential for development, redevelopment and densification to significantly increase housing and employment. **Regeneration areas** are the 20 percent most deprived areas as measured by the Government's Index of Deprivation. They require regeneration to address social exclusion allowing links to be forged with nearby opportunity areas to take advantage of their provision. **Metropolitan centres** offer the best public transport accessibility outside of central London and are therefore key to integrating land use, including housing and employment, with transport to meet sustainability objectives. **Thames Gateway** is the portion of the Thames Gateway which is situated within the GLA boundary. Policy for this region reflects Government strategy for regeneration, infrastructure improvement and development of the gateway region.

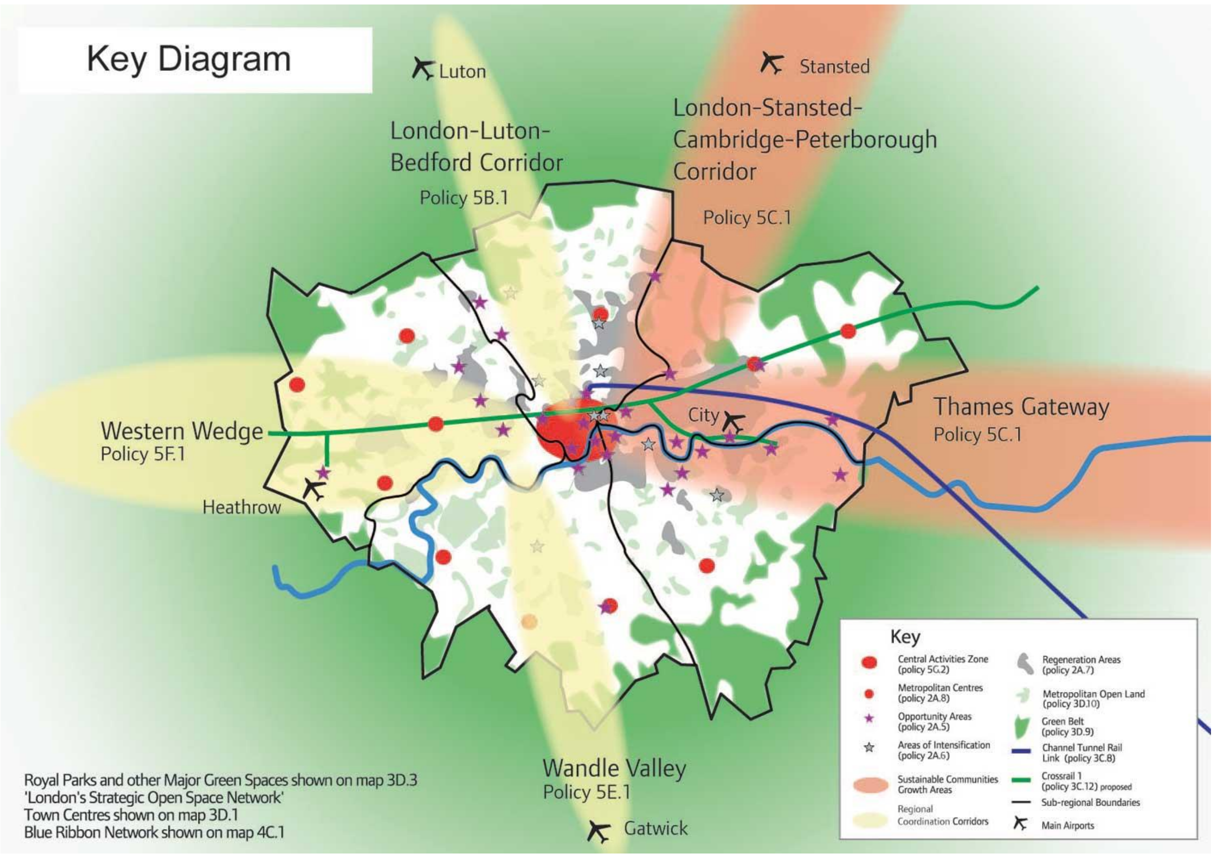

Figure 7.1: Key Diagram from the London Plan 2008 (GLA, 2008).

Green belt areas maintain the separation of towns and therefore constrain urban sprawl. Protection of the green belt should be maintained by preventing inappropriate development unless there are exceptional circumstances. **Metropolitan open land** areas consist of open space within the city, often linked by footpaths to form green chains. The preservation of open space within the built urban fabric should be maintained by providing these areas with the same level of protection as the green belt.

7.3 Spatial Interaction Model Parameterisation

In this case study the GLA region is divided into 633 zones as defined by the UK Office for National Statistics (ONS) Census Area Statistics (CAS) wards which cover the 33 London Boroughs (figure 7.2).

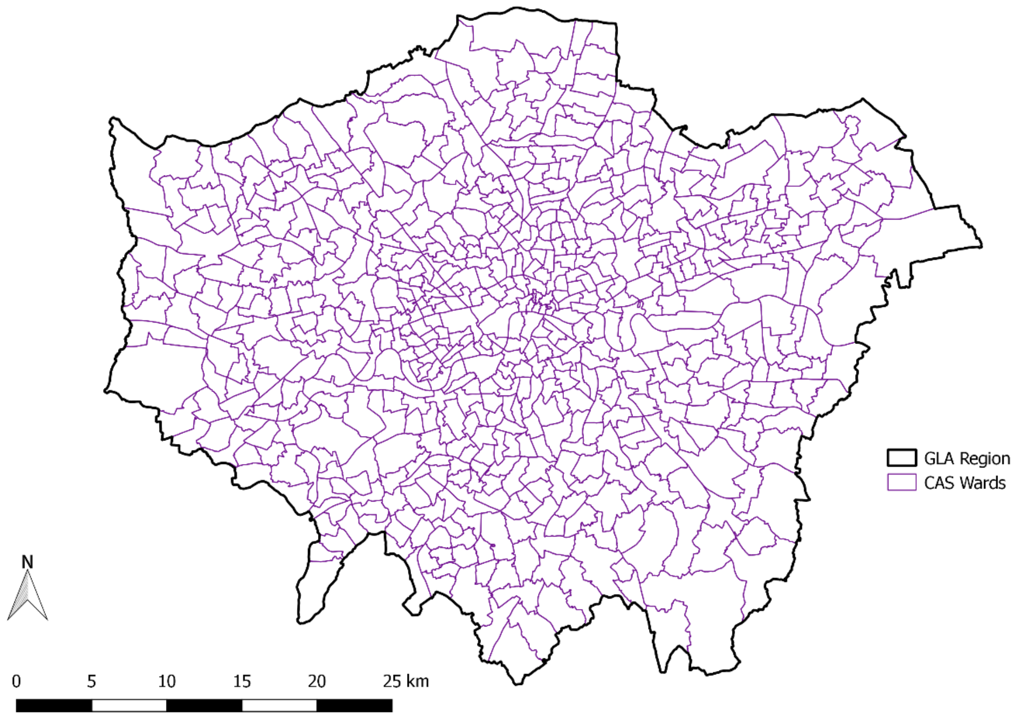

Figure 7.2: The GLA region divided into 633 zones.

Projected employment values in decade increments up to 2100 are generated exogenously by a Multi-Sectoral Dynamic Model (MDM-E3) which combines sectoral econometric models with input-output models driven by a Global energy-environment-economy model (E3MG). MDM-E3 employs a 42-sector UK Standard Industrial Classification (SIC) which, in this case study, is aggregated to employment in five industrial sectors; Primary industries, Retail, Construction, Finance, and Other services (for example public sector) as shown in figure 7.3.

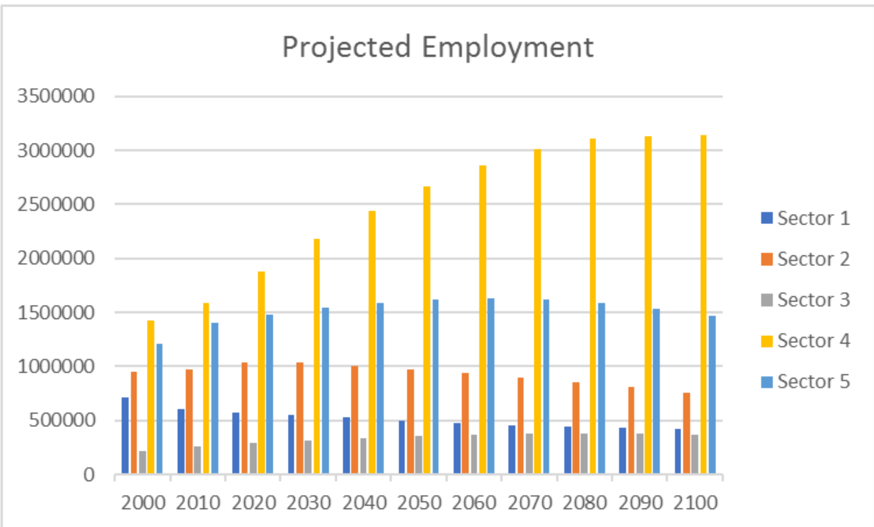

Figure 7.3: MDM-3 employment projection aggregated to 5 sectors.

The key diagram attractors described in section 7.2.1 are used in the spatial disaggregation of both employment and population along with their respective observed values; the use of observed employment and population as attractors reflects the tendency for new development to agglomerate around existing development. Areas of opportunity, regeneration, and the Thames Gateway occupy large tracts of overlapping land as shown in figure 7.4. Metropolitan centres and intensification areas are localised into small pockets of land distributed around the GLA region as shown in figure 7.5 along with previously developed land; the London Plan sets out a policy for the use of **previously developed land** consisting of Brownfield sites whose development should be maximised as part of a wider set of sustainability criteria. These spatial attractors are shown in table 7.1.

Attractor: zonal attribute or area within each zone	Weight
Brownfield	1
Gateway	1
Intensification	1
Metropolitan	1
Opportunity	1
Regeneration	1
Observed Employment / Population	1

Table 7.1: SIM Employment and Population Attractors

The key diagram constraints described in section 7.2.2 are used to prevent inappropriate development; protected areas of metropolitan land and the green belt are shown in figure 7.6. Additionally, the London Plan identifies the following areas, shown in figure 7.7, which are used as development constraints:

- Special Areas for Conservation.
- Sites of Special Scientific Interest (SSSI).
- Local Nature Reserves.
- National Nature Reserves.

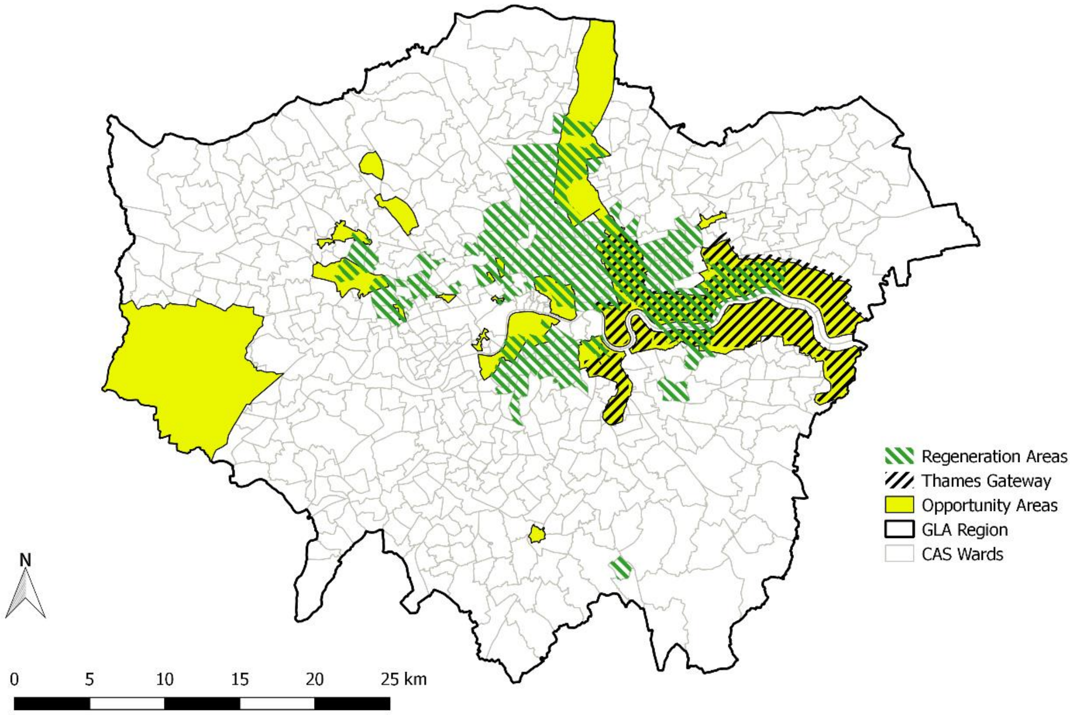

Figure 7.4: Case Study Attractors; Opportunity, Regeneration and Thames Gateway.

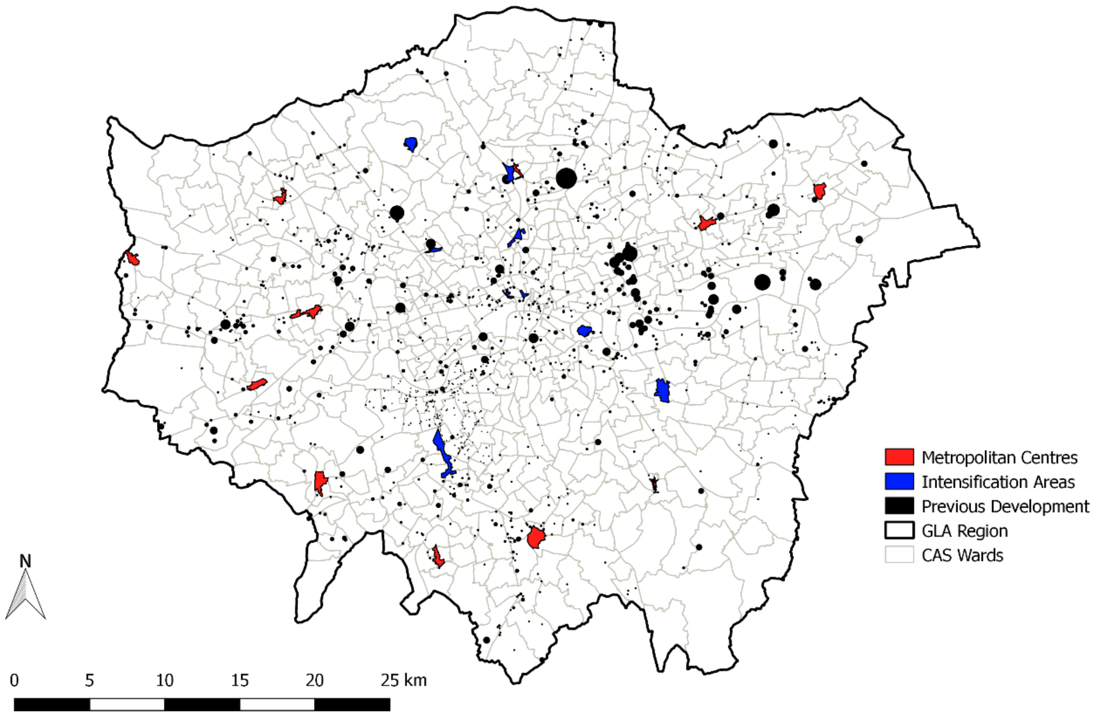

Figure 7.5: Case Study Attractors; Metropolitan centres, Intensification and Previous Development.

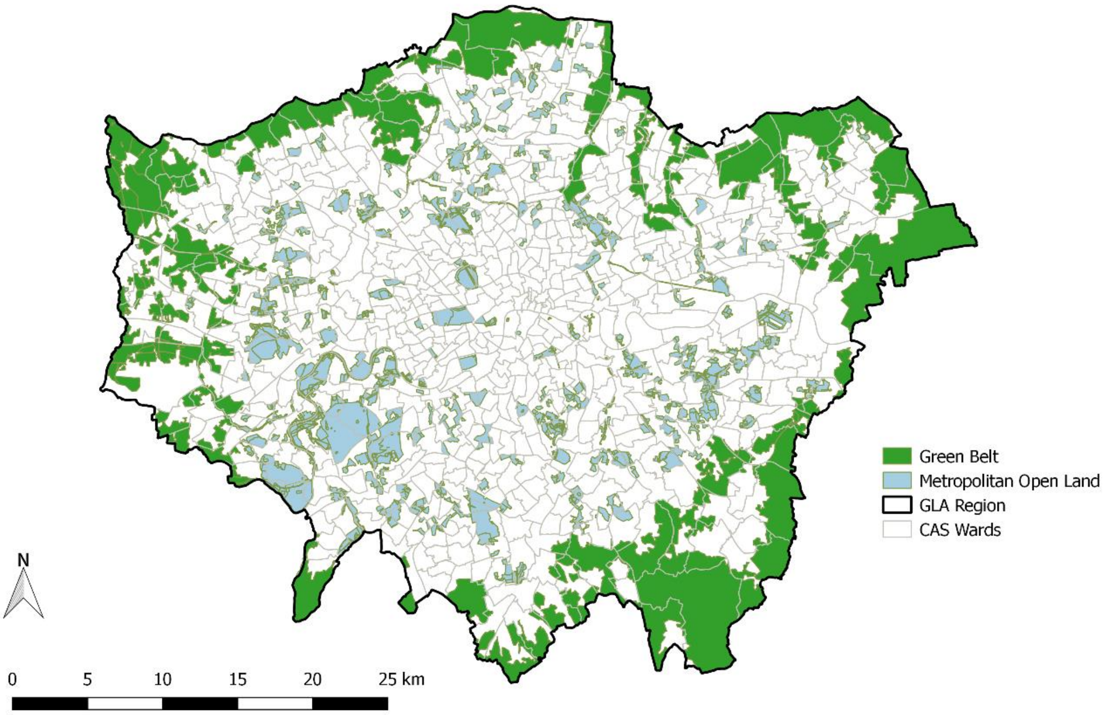

Figure 7.6: Case Study Constraints; Greenbelt and Metropolitan Open Land.

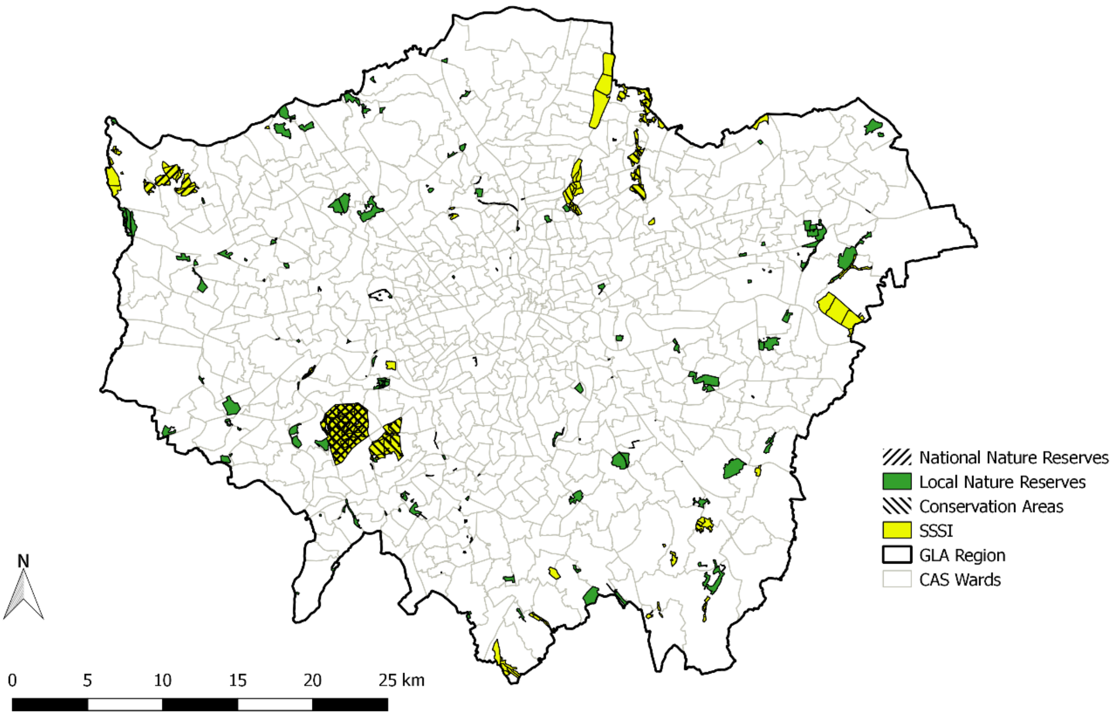

Figure 7.7: Case Study Constraints: Nature, Conservation and SSSI.

The Spatial Interaction Model (SIM) does not redevelop existing areas so currently developed land and bodies of water, as shown in figure 7.8, are used as practical constraints on future development. To ensure consistency across modelling scales the available land within each zone in the SIM is computed from the input constraint raster used in the Urban Development Model (UDM). The constraints described in this section are combined into a single vector layer which is rasterised with a cell size of 1m before being down-sampled to a raster with a cell size of 100m using the framework function $BooleanDownsampler()$ with a threshold of 50%. The framework function $AreaFromRaster()$ is then used to output a .csv file containing an available area value for each zone computed from the number of free cells multiplied by the cell size squared.

Observed data for initial employment in each employment sector for each zone (shown in figure 7.9 aggregated over all sectors) is provided by ONS official labour market statistics (National Online Manpower Information System - NOMIS) using Annual Business Inquiry (ABI) data for 2005, whilst observed data for initial population in each zone (shown in figure 7.11) is provided by ONS England and Wales Census counts for 2001. The ONS Generalised Land Use Database (GLUD) classifies land use in the Ordnance Survey (OS) MasterMap Topography Layer and documents the total area occupied by each category. The area occupied by domestic buildings is used along with observed population to compute the observed population density for each zone (shown in figure 7.12) whilst the area occupied by non-domestic buildings is used along with observed employment to compute the observed employment density for each zone as shown in figure 7.10.

A pragmatic check when running the SIM showed that the projected values of employment and population could not be housed for each timestep using their respective observed density values. One option to remedy this situation would be to remove the greenbelt constraint freeing up development area but the London Plan stipulates that this should not be done unless there are exceptional circumstances. For the purposes of this case study, the observed density values for each zone were doubled which allows new development, for both employment and population, to be housed in each timestep. This rather crude approach will be refined and presented as a densification scenario in the discussion chapter.

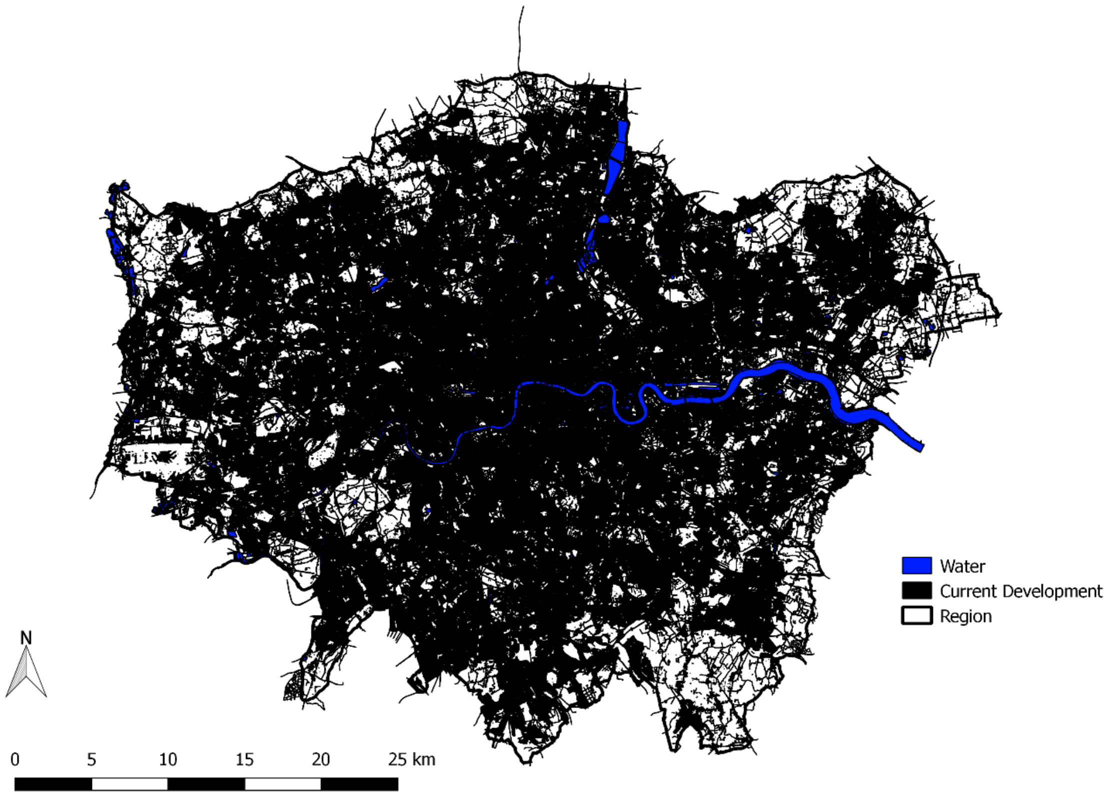

Figure 7.8: Case Study Constraints: Current Development and Water.

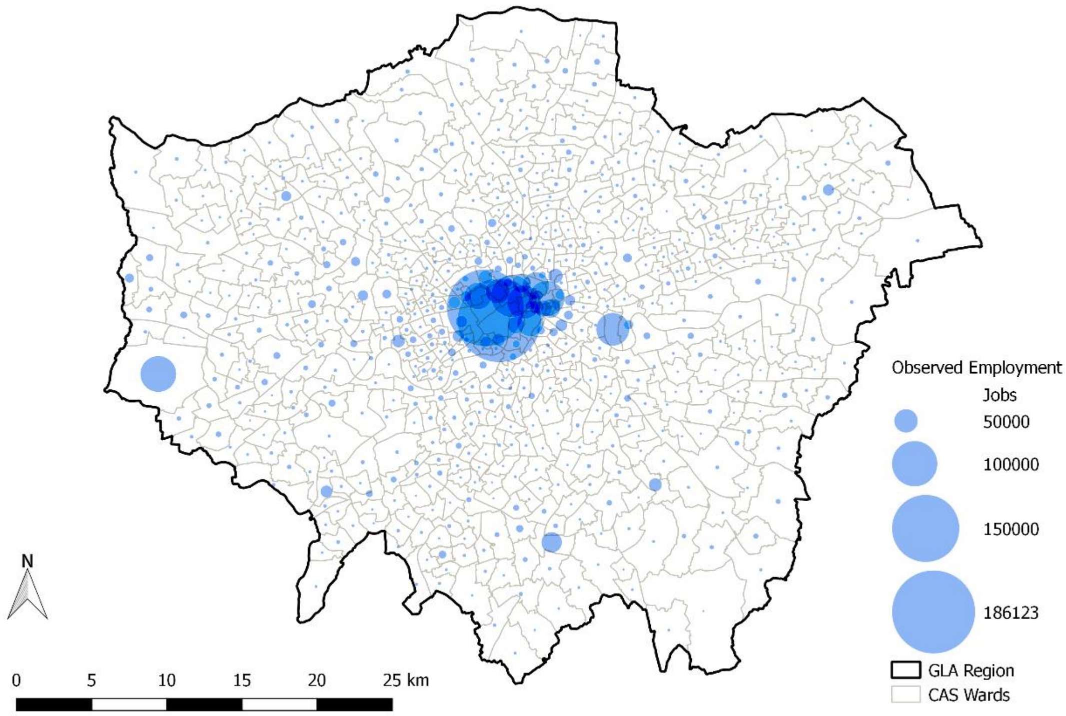

Figure 7.9: Observed Employment.

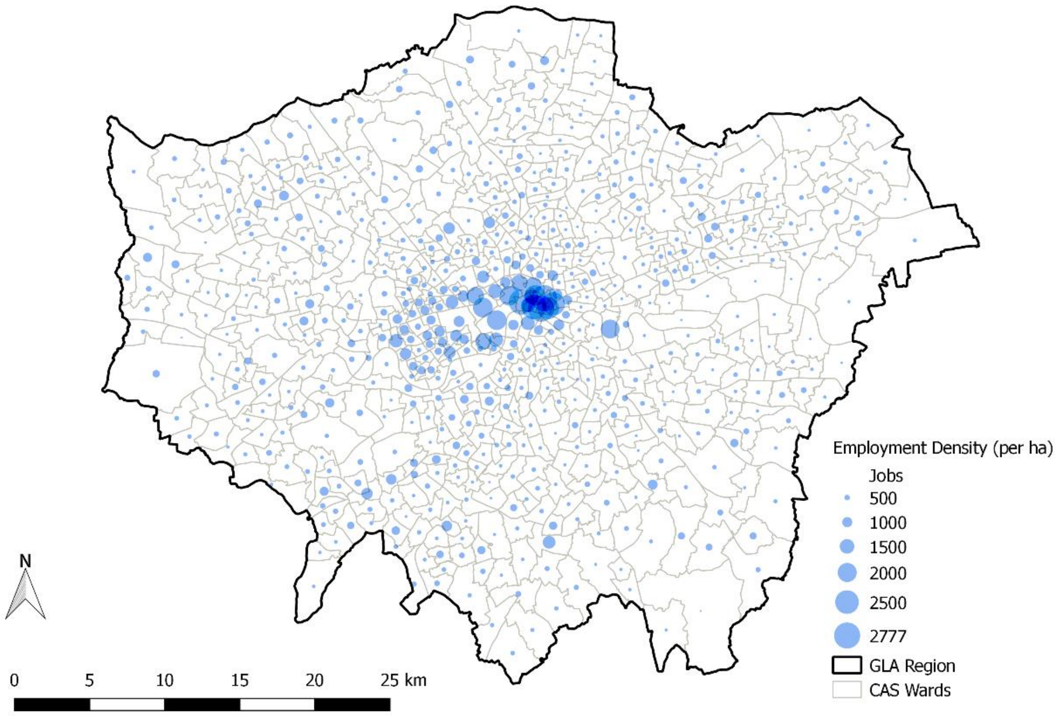

Figure 7.10: Employment Density.

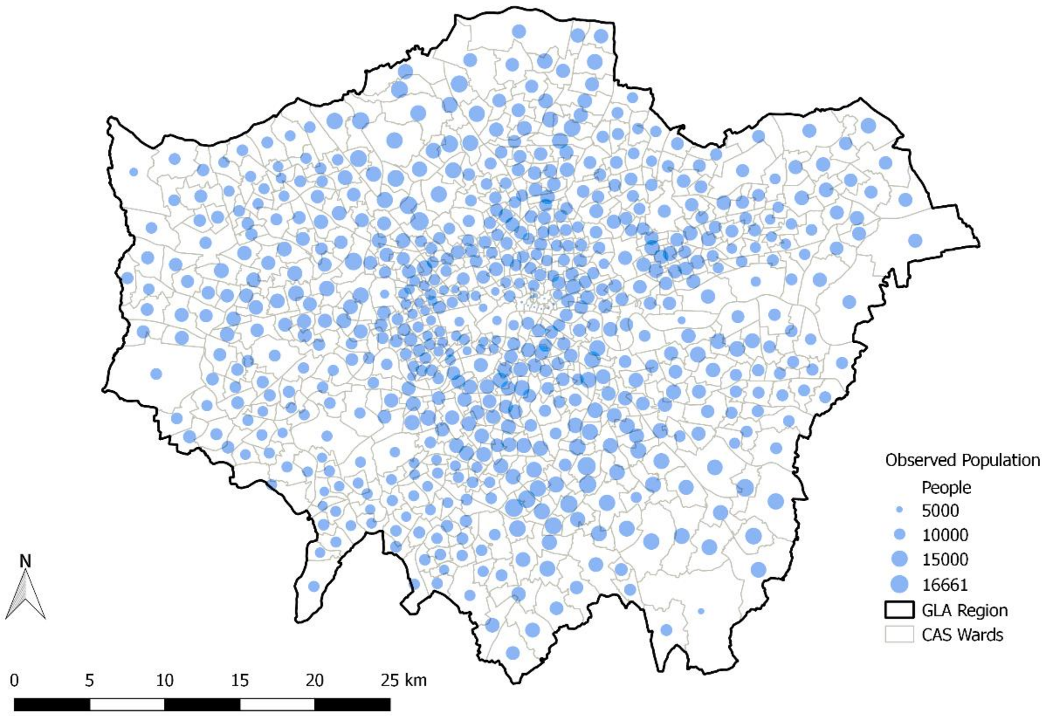

Figure 7.11: Observed Population.

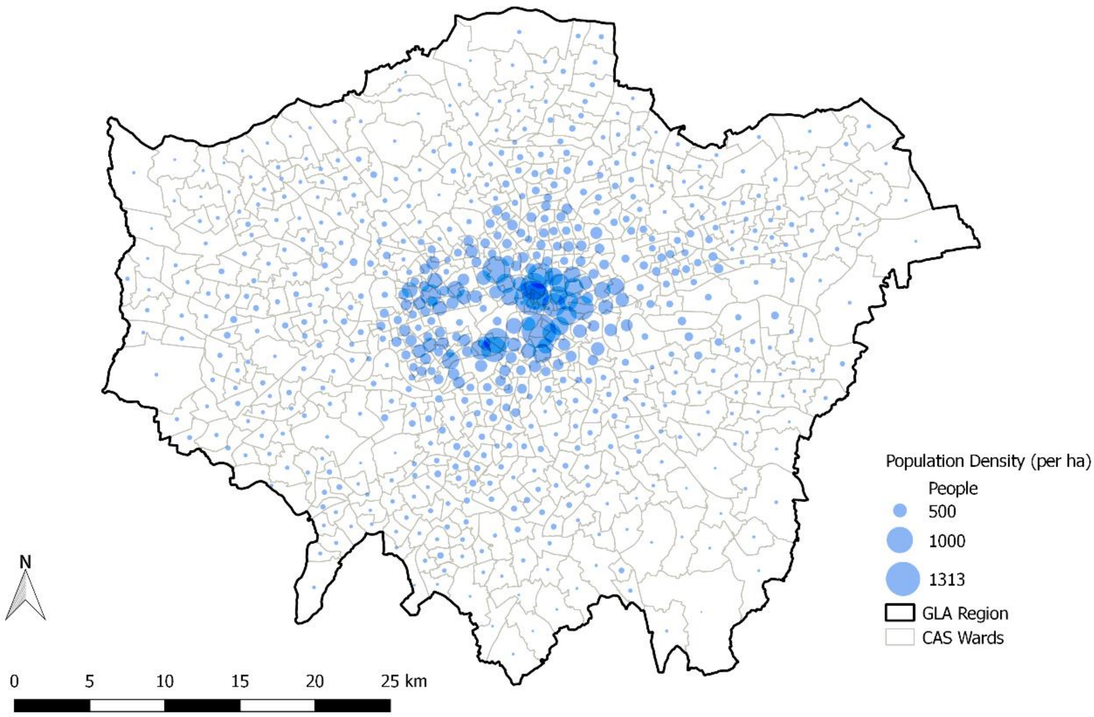

Figure 7.12: Population Density.

7.4 Transport Accessibility Model Parameterisation

7.4.1 Road

Spatial model inputs as shown in figure 7.13 consist of zone centroids as points which act as journey origins and destinations, and the network of major roads as edges along which journeys run in the GLA region; the 'Congestion Charge Zone' polygon with the western extension removed i.e. post-2011, is included to implement route-dependent charging which results in journeys being routed to avoid the charge where a lower cost network path exists. Key generalised cost parameters for the road network are shown in table 7.2.

Spatial Data	**Key Parameter Values**
OS Mastermap Integrated Transport Network (ITN)	Walking speed: 6 km/h. Access time: 3 minutes. Vehicle occupancy: 1.16 people per trip (WebTAG, 2009). Value of time: 1 hour = £5.04 (WebTAG, 2009). Congestion charge: £8 for journeys entering or leaving the congestion zone, 90% discount applied to routes originating within the zone (TFL, 2013).

	Operating costs (fuel): computed using the coefficients a, b, c and d (WebTAG, 2009). $a = 88.205$ $b = 5.727$ $c = -0.02865$ $d = 0.00034$ Operating costs (non-fuel): computed using the coefficient a (WebTAG, 2009). $a = 3.846$ Travel speed: average speed for three cordons in the city from 2006 London Travel Report (TFL 2006).

Table 7.2: Key generalised cost parameters for the road network

7.4.2 Bus

Spatial model inputs as shown in figure 7.14 consist of bus stops as points and the network of connecting roads as edges; a vector of the GLA region containing zone polygons is provided to configure access edges from zone centroids to nearest stops. Key generalised cost parameters for the bus network are shown in table 7.3.

Spatial Data	**Key Parameter Values**
Transport for London data produced by Jacobs Consulting	Walking speed: 6 km/h. Interchange distance: 640m (TFL, 2010). Walking disincentive: 1.6 (WebTAG, 2009). Waiting disincentive: 2.6 (WebTAG, 2009). Fare: flat rate of £1 assuming fixed cost of Oyster card as used in over 85% of journeys within London (TFL, 2007). Value of time: 1 hour = £5.04 (WebTAG, 2009). Average wait: half the average frequency of service (assumed to be 3 minutes). Travel speed: equivalent edge speed computed from travel time included in dataset.

Table 7.3: Key generalised cost parameters for the bus network

7.4.3 Light Rail

Spatial model inputs as shown in figure 7.15 consist of light rail stations as points and the network of connecting tracks as edges; a vector of the GLA region containing zone polygons is provided to configure access edges from zone centroids to nearest stations. Key generalised cost parameters for the light rail network are shown in table 7.4.

Spatial Data	**Key Parameter Values**
OS Meridian data with all links assumed to be bi-directional and passenger-carrying	Walking speed: 6 km/h. Interchange distance: 960m (TFL, 2010). Walking disincentive: 1.6 (WebTAG, 2009). Waiting disincentive: 2.6 (WebTAG, 2009). Fare: average of £0.18 per km (TFL, 2006). Value of time: 1 hour = £5.04 (WebTAG, 2009). Average wait: half the average frequency of service (assumed to be 3 minutes). Travel speed: equivalent average speeds computed from timetabled sections of light rail network in the GLA area.

Table 7.4: Key generalised cost parameters for the light rail network

7.4.4 Heavy Rail

Spatial model inputs as shown in figure 7.16 consist of heavy rail stations as points and the network of connecting tracks as edges; a vector of the GLA region containing zone polygons is provided to configure access edges from zone centroids to nearest stations. Key generalised cost parameters for the heavy rail network are shown in table 7.5.

Spatial Data	**Key Parameter Values**
OS Meridian data with all links assumed to be bi-directional and passenger-carrying	Walking speed: 6 km/h. Interchange distance: 960m (TFL, 2010). Walking disincentive: 1.6 (WebTAG, 2009). Waiting disincentive: 2.6 (WebTAG, 2009). Fare: average of £0.18 per km (TFL, 2006). Value of time: 1 hour = £5.04 (WebTAG, 2009). Average wait: half the average frequency of service (assumed to be 7.5 minutes).

	Travel speed: equivalent average speeds computed from timetabled sections of heavy rail network in the GLA area.

Table 7.5: Key generalised cost parameters for the heavy rail network

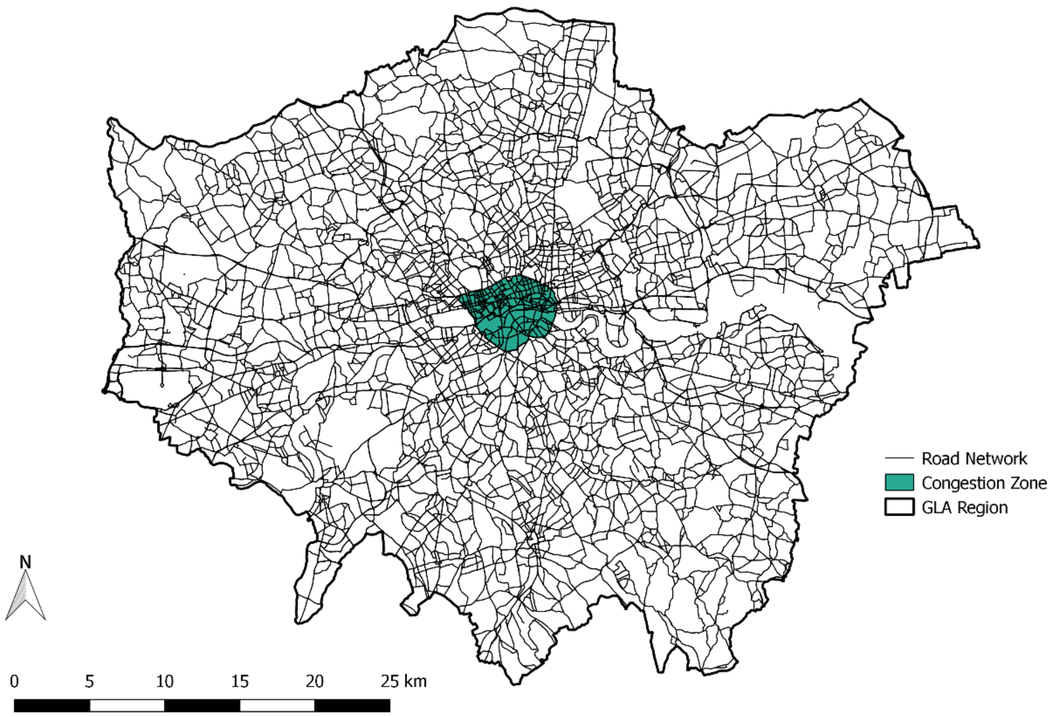

Figure 7.13: Road Network and Congestion Zone.

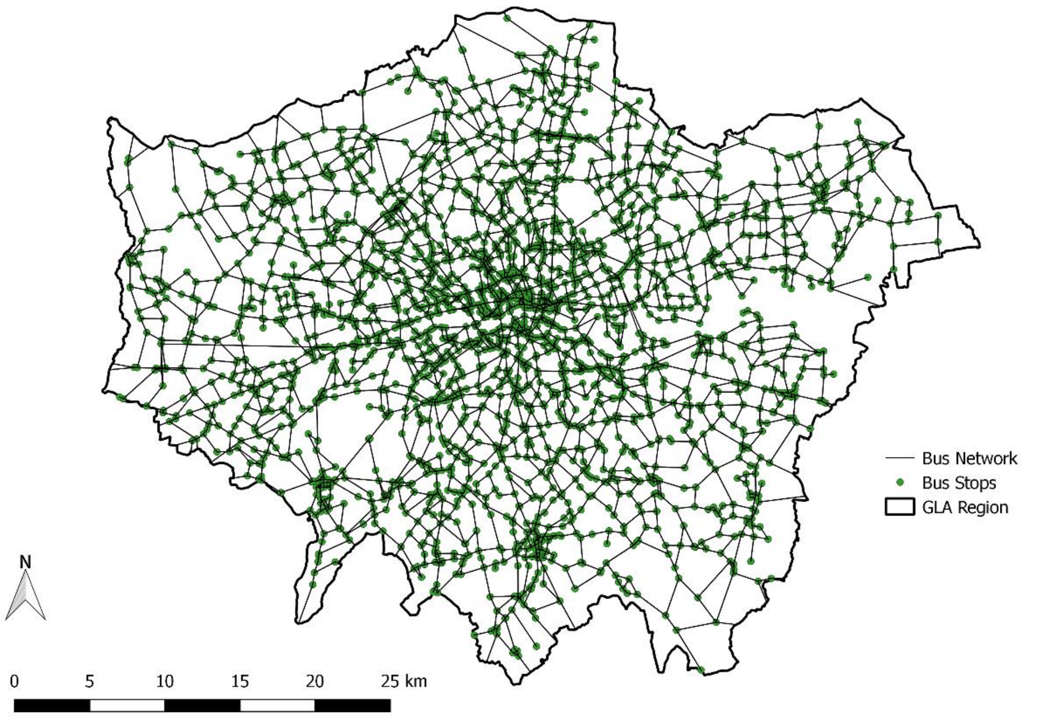

Figure 7.14: Bus Network and Stops.

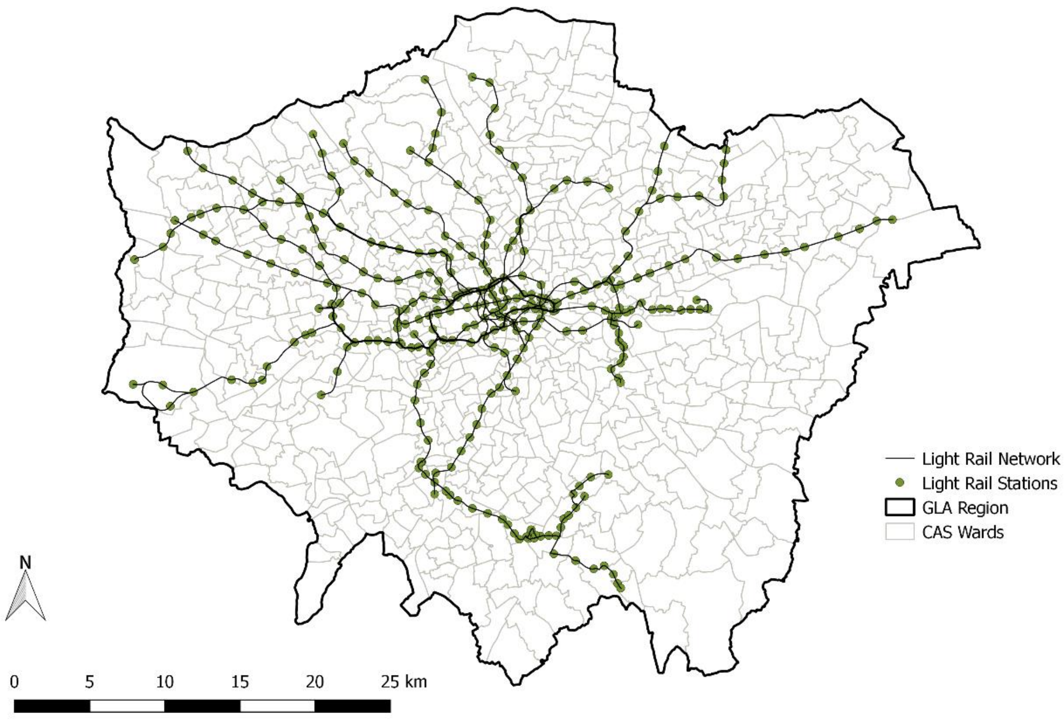

Figure 7.15: Light Rail Network and Stations.

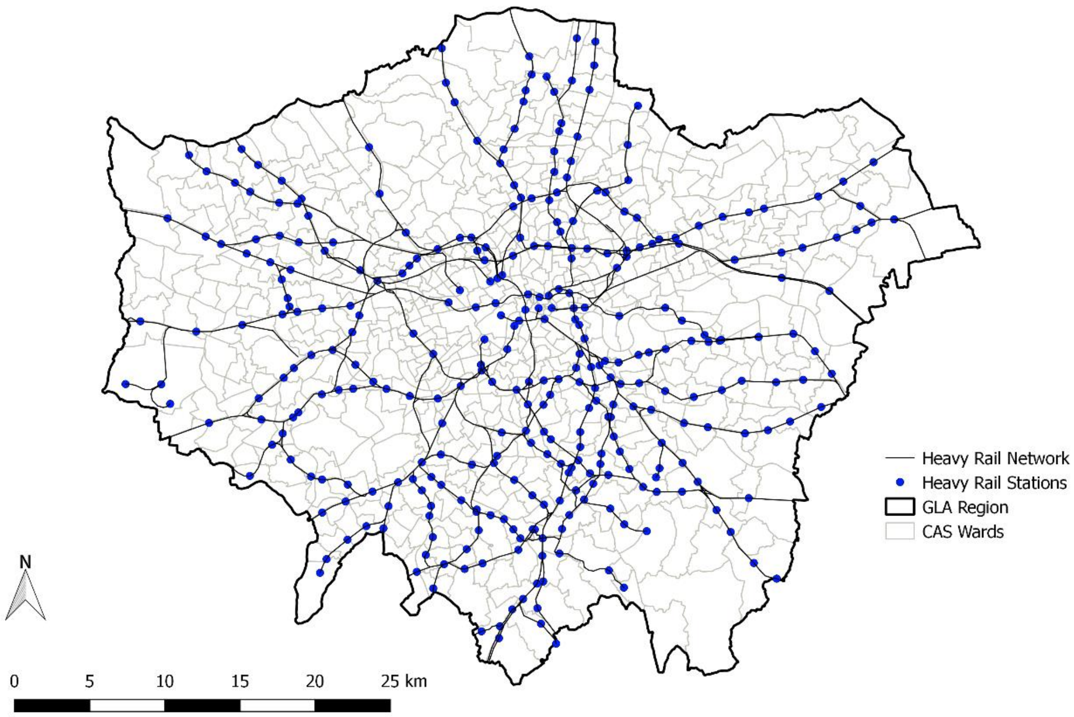

Figure 7.16: Heavy Rail Network and Stations.

7.5 Urban Development Model Parameterisation

A raster of integer zone identity values must be created for the UDM to correctly assign cells to zones. These values range from zero to the total number of zones minus one and correspond to the index to the zone array used in the UDM.

To ensure consistency across modelling scales the London Plan attractors used in the SIM for inter-zonal population disaggregation are used in the UDM for intra-zonal population disaggregation at the cellular level. Areas of opportunity, regeneration, and intensification, along with metropolitan centres and the Thames Gateway are merged into a single vector layer before converting to a Boolean raster of London Plan suitability as shown in figure 7.17; previously developed land is also converted to a Boolean suitability raster as shown in figure 7.18. Inter-zonal population disaggregation is also driven by accessibility to employment via the Transport Accessibility Model (TAM) so intra-zonal population disaggregation employs localised accessibility to transportation; rasters of proximity to the road network (shown in figure 7.20), and proximity to public transport access points (shown in figure 7.21) are used for this purpose. The tendency for new development to agglomerate around existing development is also represented at the intra-zonal level by using rasterised proximity to current development as shown in figure 7.19. These spatial attractors are shown in table 7.6.

Attractor	**Weight**
London plan	0.2
Previous development	0.2
Development proximity	0.2
Road proximity	0.2
Public transport proximity	0.2

Table 7.6: UDM Spatial Attractors and Weights

As well as using consistent attractors, coherence across modelling scales is ensured by using consistent constraints which are used to define the available area for new development in both the SIM and UDM. The constraints used by the SIM as shown in figures 7.6, 7.7 and 7.8 are merged into a single vector layer and rasterised as described in section 7.3 to create a Boolean raster of development constraint (shown in figure 7.22).

7.6 Framework Setup

A shown in figure 7.23, 11 models are required to model the GLA region; a network processing model and generalised cost model for each of the 4 input transport networks respectively, followed by an accessibility model to aggregate and convert input cost matrices. The output accessibility matrix is fed into the SIM which, in turn, generates zonal population data to be used by the UDM.

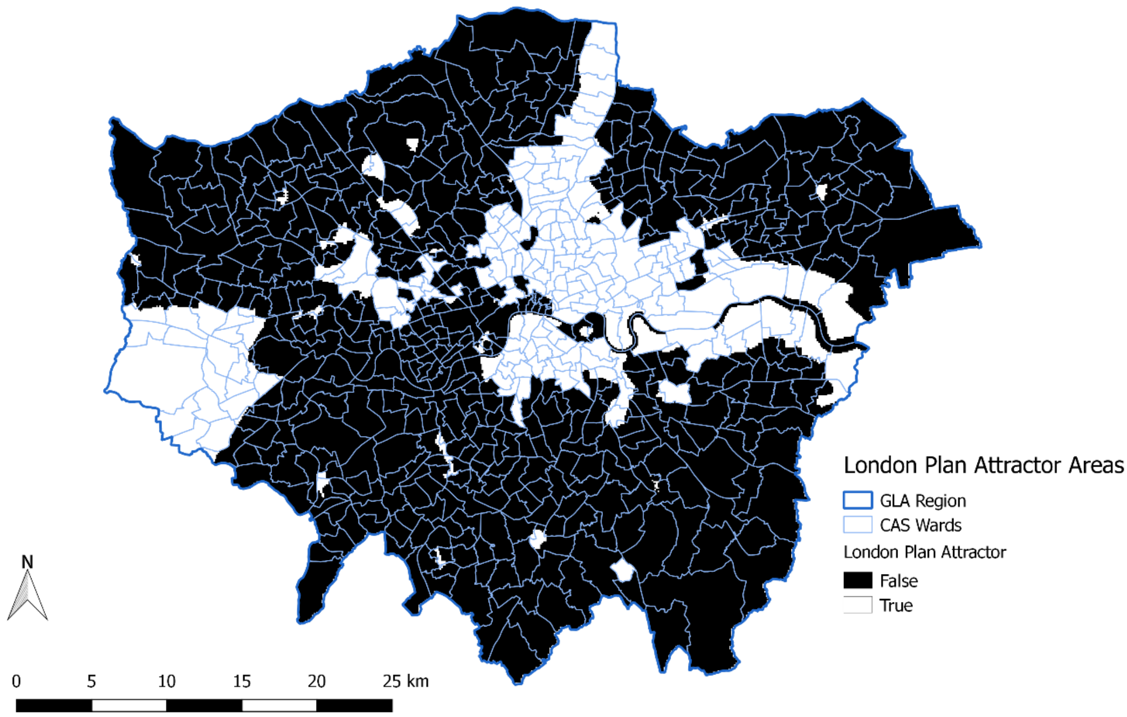

Figure 7.17: London Plan Suitability Raster.

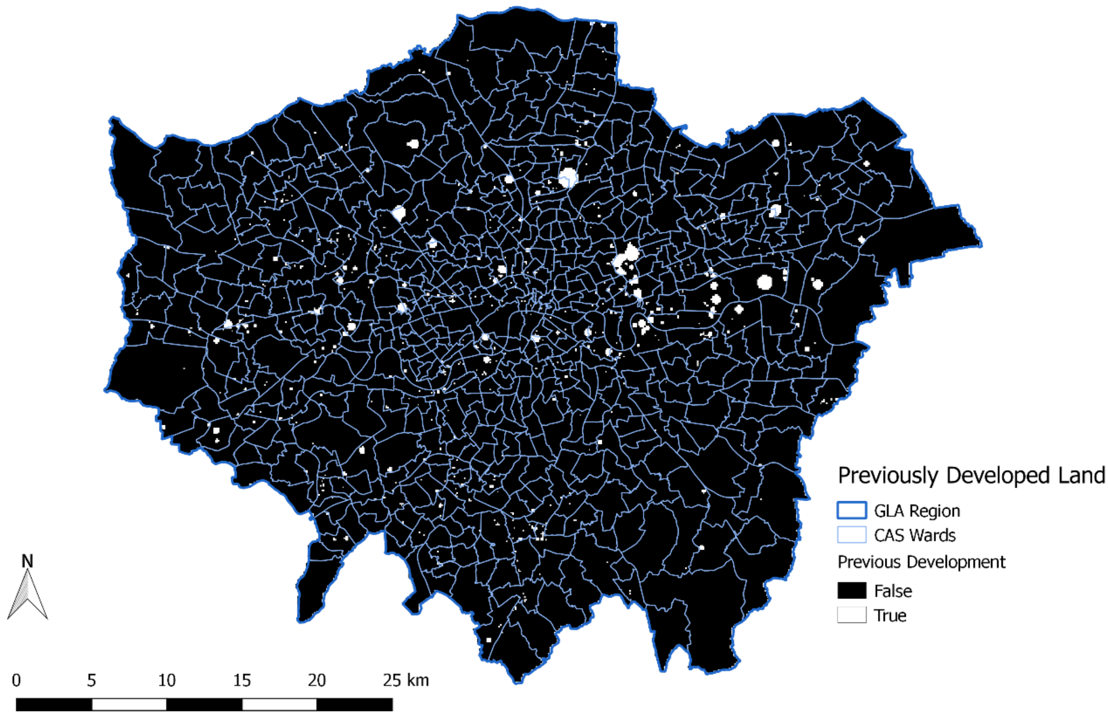

Figure 7.18: Previous Development Suitability Raster.

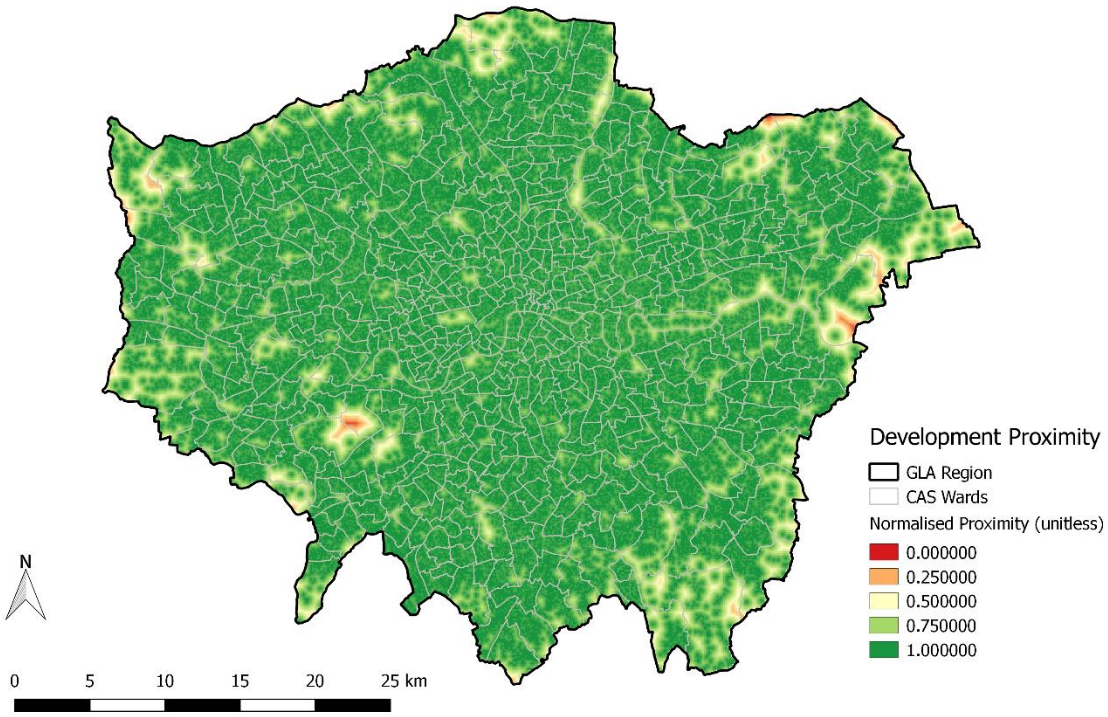

Figure 7.19: Current Development Proximity Raster.

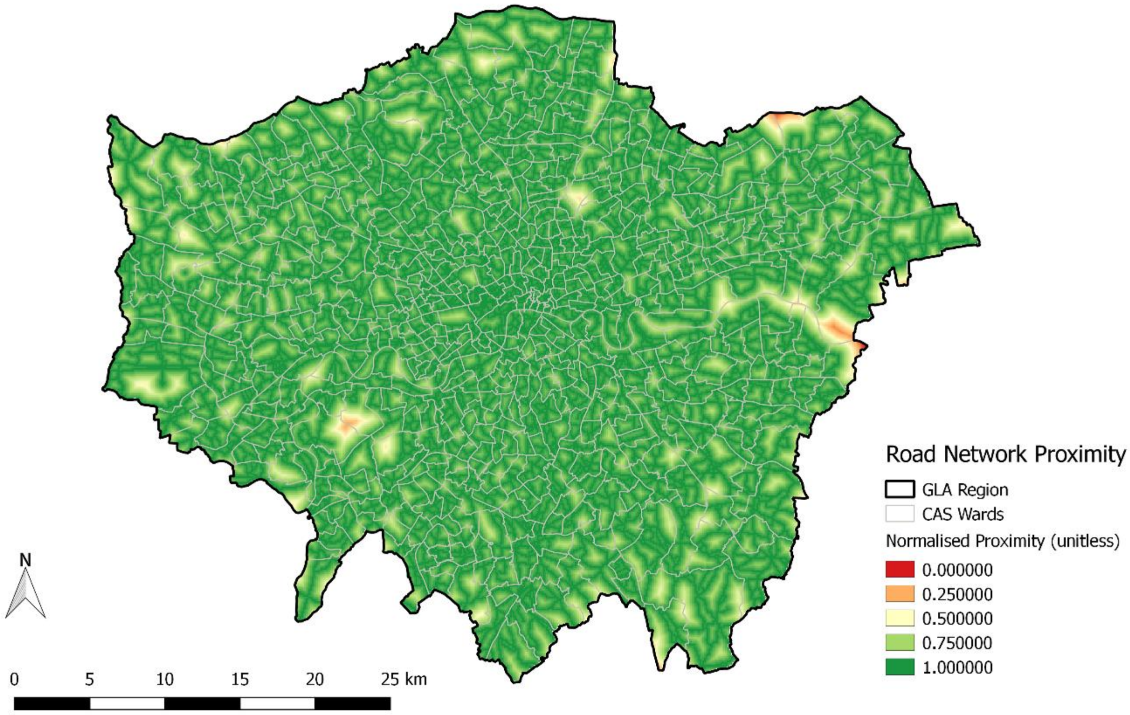

Figure 7.20: Road Network Proximity Raster.

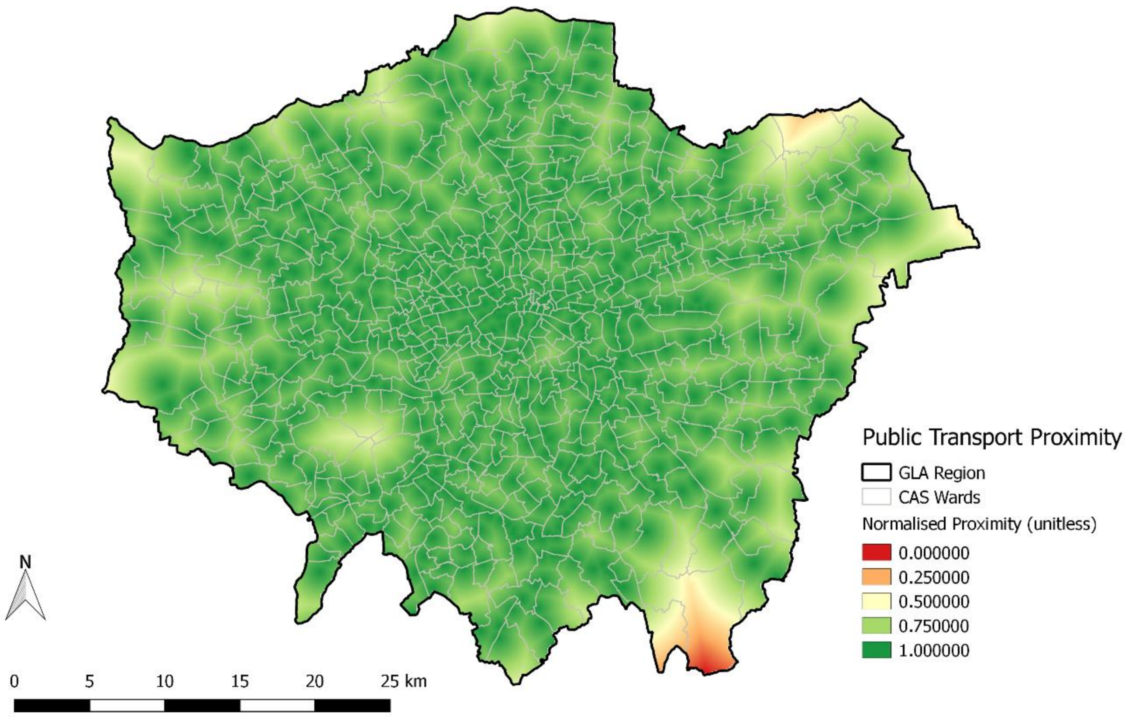

Figure 7.21: Public Transport Proximity Raster.

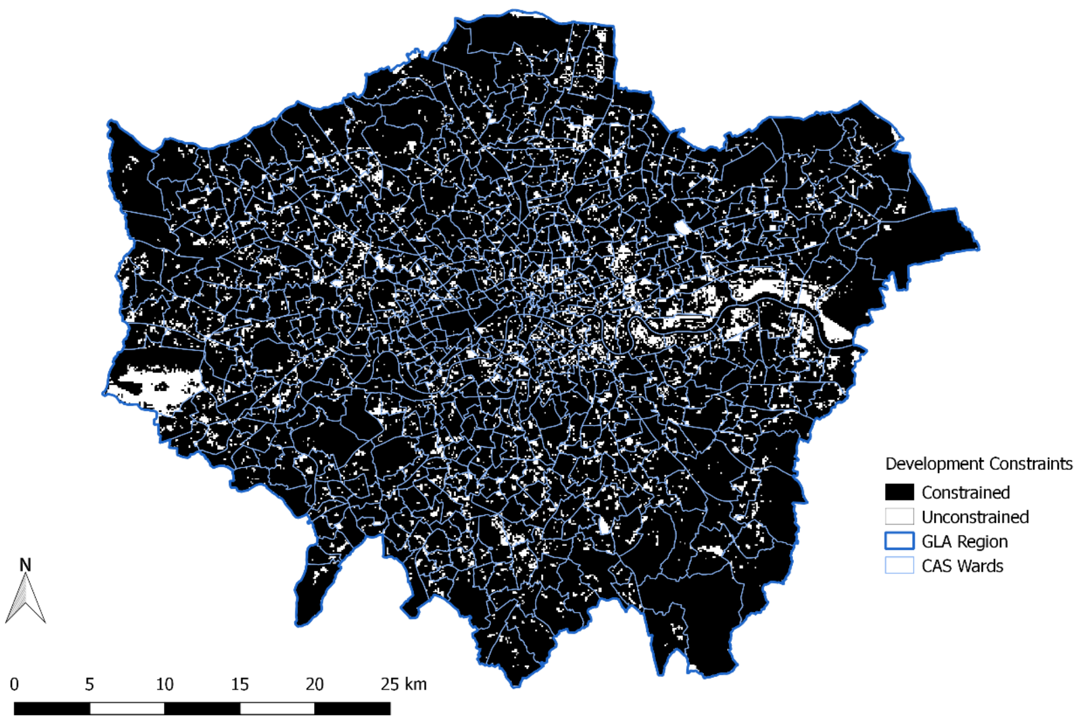

Figure 7.22: Development Constraint Raster.

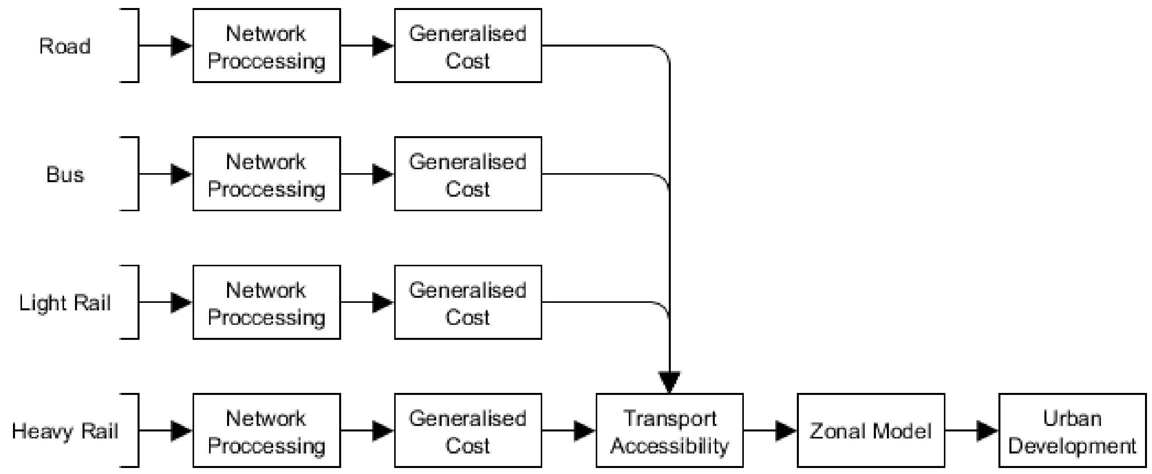

Figure 7.23: Sequence of models used to simulate the GLA region

As shown in figure 7.24, PostgreSQL databases are used to configure and run the collection of models for the GLA region. Model and model group data in .csv format is prepared in a spreadsheet using a template table for each dataset before loading to the $CSLon\_ModelGroups$ and $CSLon\_RunData$ databases. Raster data is prepared in $QGIS$ and exported using the asci file format before converting and loading into the $CSLon\_RunData$ database. Vector data is prepared in $QGIS$ and imported to the $CSLon\_RawData$ database using the shapefile import/export plugin in $pgadmin$. This vector data is then processed to extract the required columns of data, reformatting where necessary and loading into tables in the $CSLon\_RunData$ database

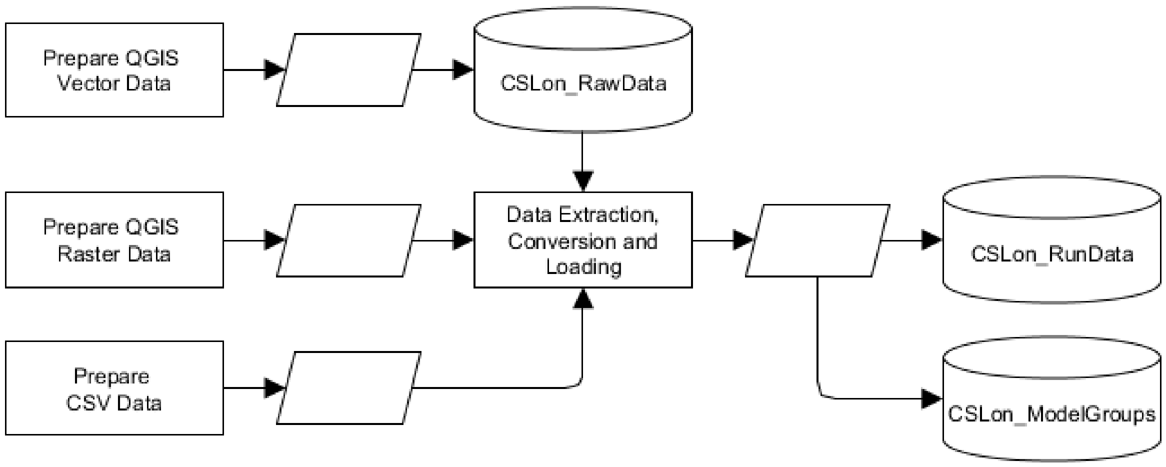

Figure 7.24: Model configuration process for the GLA region.

7.7 Results

7.7.1 Transport Accessibility Model

The output from the road network processing model adds access edges linking zone centroid origins and destinations to the input road network as shown in figure 7.25. In addition, edges which span the congestion charge boundary are identified as shown in figure 7.26 so that the provided congestion charge value can be used as an edge weight in the computation of least-cost paths. The cost matrix output from the road network generalised cost model (figure 7.27) shows a radial spread of cost values like that which would be obtained by using the Euclidean distance between zone origins.

The output from the bus network processing model adds access edges linking zone centroid origins and destinations with one or more nearest bus stops to the input bus network as shown in figure 7.28. In addition, interchange edges are created between stops within the defined radius of one another as shown in figure 7.29. The cost matrix output from the bus network generalised cost model (figure 7.30) shows an approximately radial spread of cost values like that obtained using the shared network of major roads but with high-costs evident where access edges are significant resulting in long access times to nearest stops.

The output from the light rail network processing model adds access edges linking zone centroid origins and destinations with one or more nearest light rail stations to the input light rail network as shown in figure 7.31. In addition, interchange edges are created between stations within the defined radius of one another as shown in figure 7.32. The cost matrix output from the light rail network generalised cost model (figure 7.33) shows a spread of cost values where low costs follow the network, corresponding to zones with nearby access to stations, and high costs represent the time and disincentive to walk to the nearest station.

The output from the heavy rail network processing model adds access edges linking zone centroid origins and destinations with one or more nearest heavy rail stations to the input heavy rail network as shown in figure 7.34. In addition, interchange edges are created between stations within the defined radius of one another as shown in figure 7.35. The cost matrix output from the heavy rail network generalised cost model (figure 7.36) shows a spread of cost values where low costs follow the network, corresponding to zones with nearby access to stations, and high costs represent the time and disincentive to walk to the nearest station.

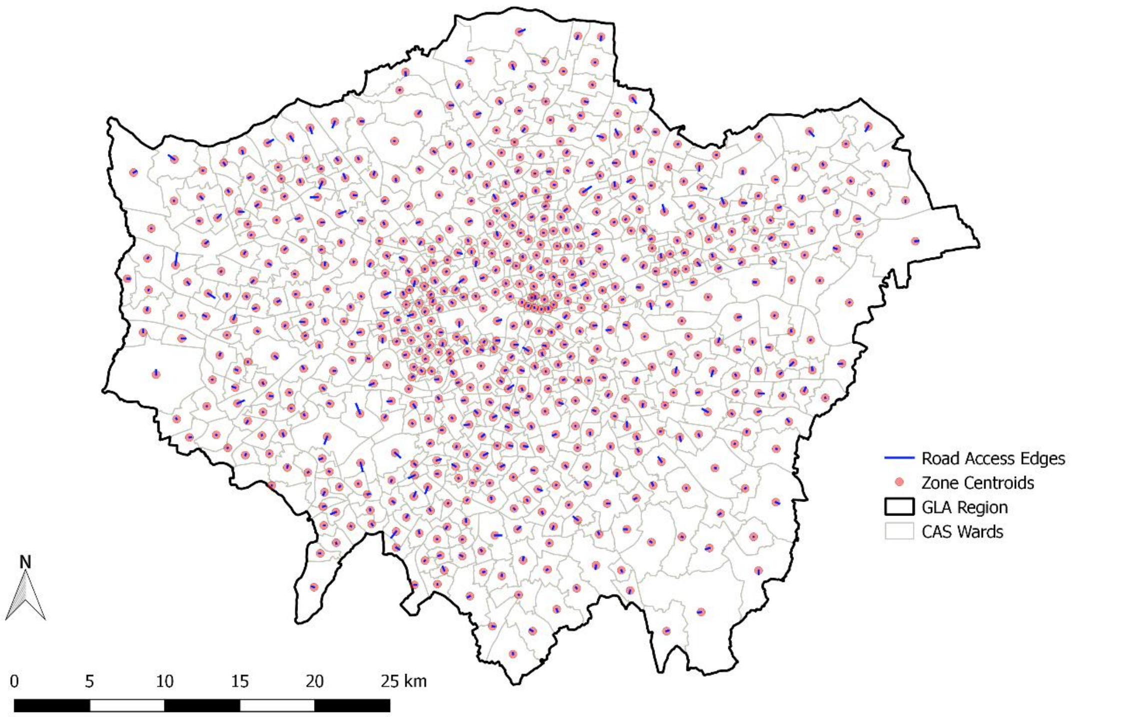

Figure 7.25: Road Network Access Edges.

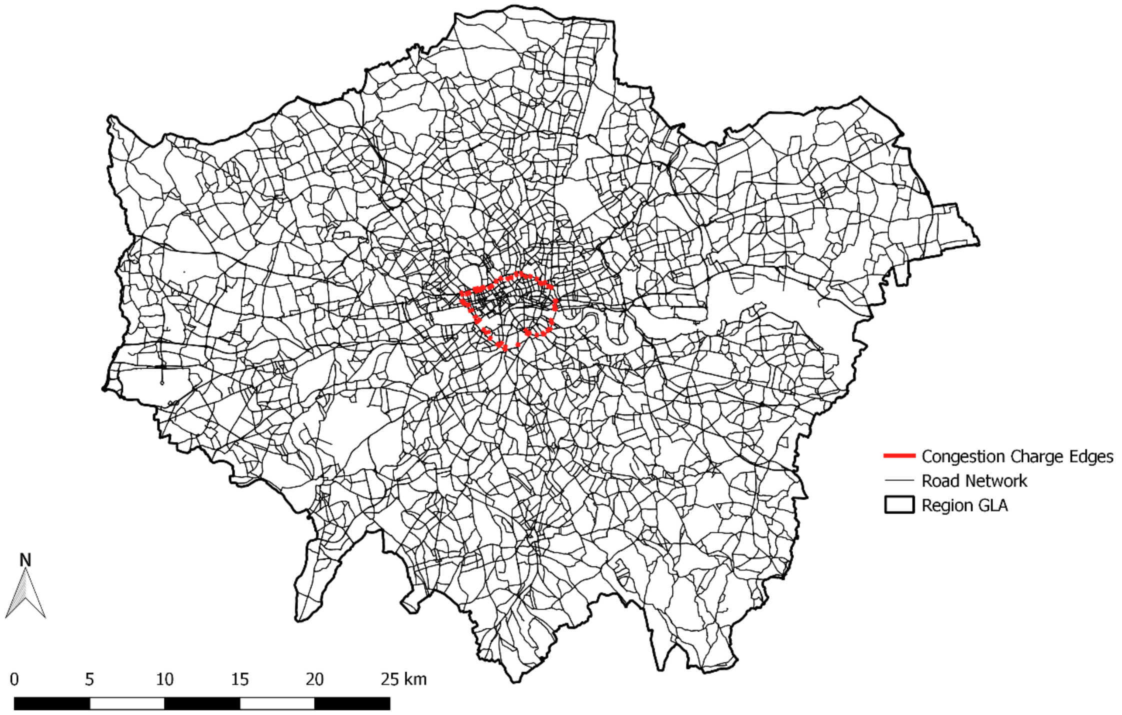

Figure 7.26: Road Network Congestion Charge Edges.

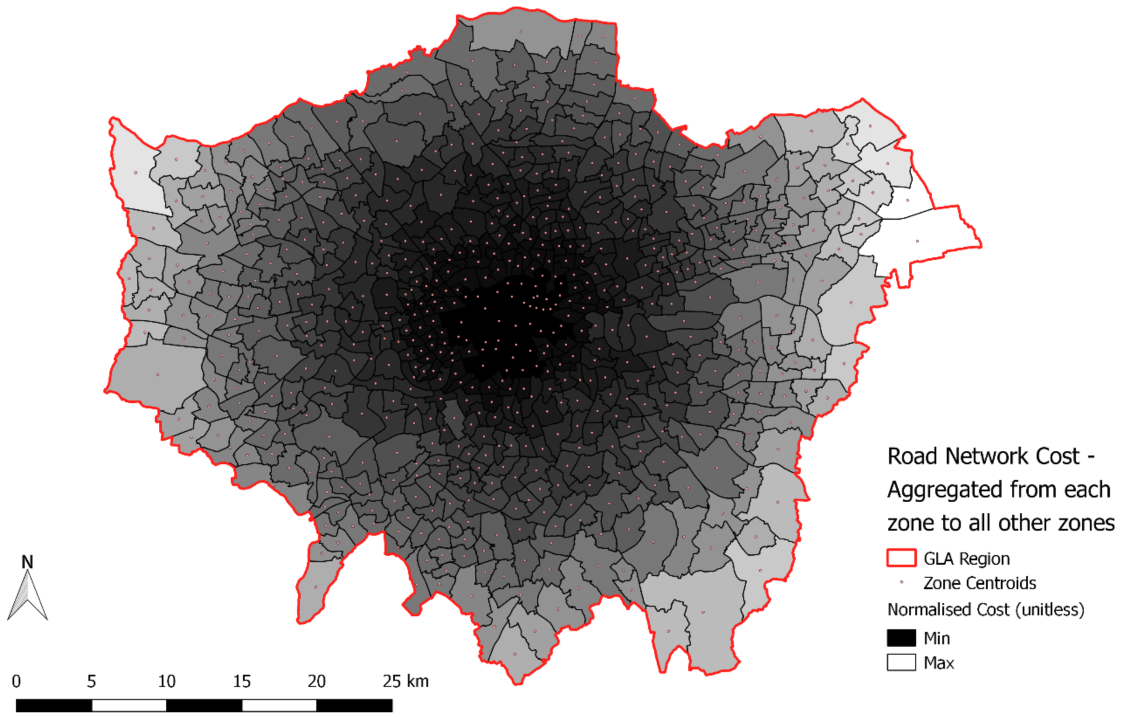

Figure 7.27: Road Network Generalised Cost.

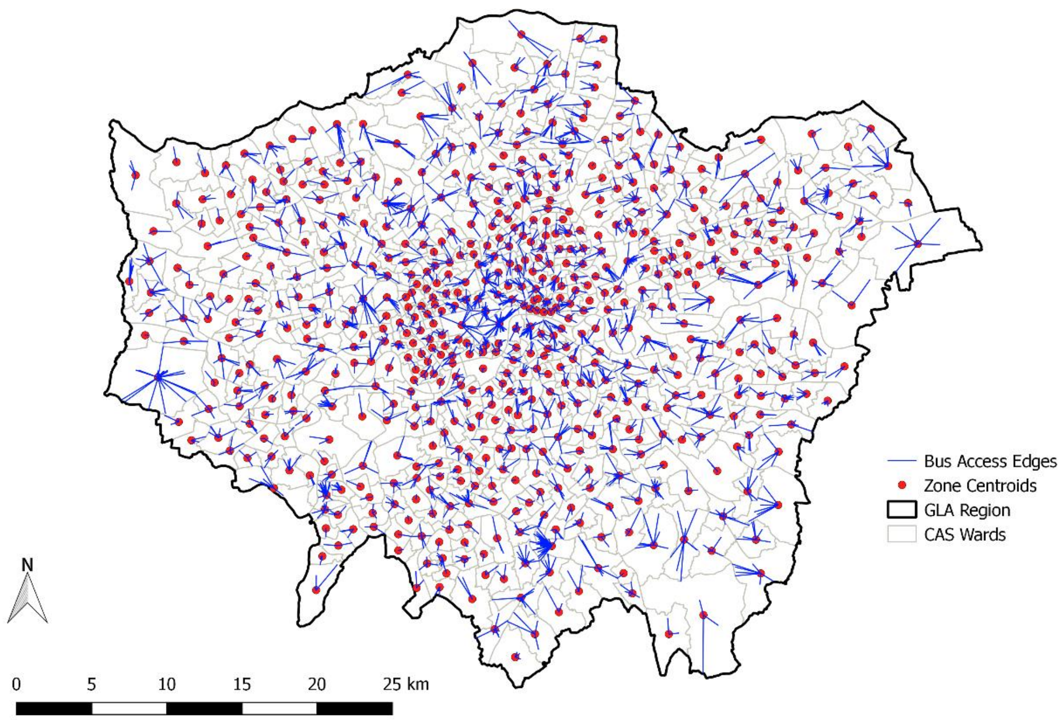

Figure 7.28: Bus Network Access Edges.

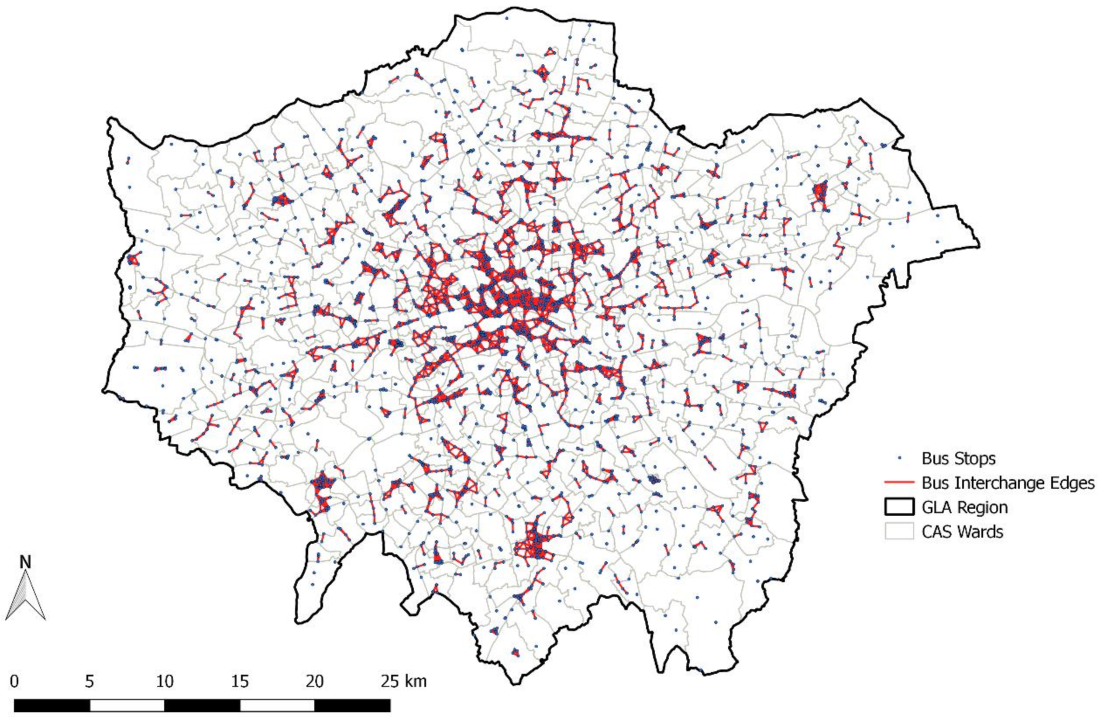

Figure 7.29: Bus Network Interchange Edges.

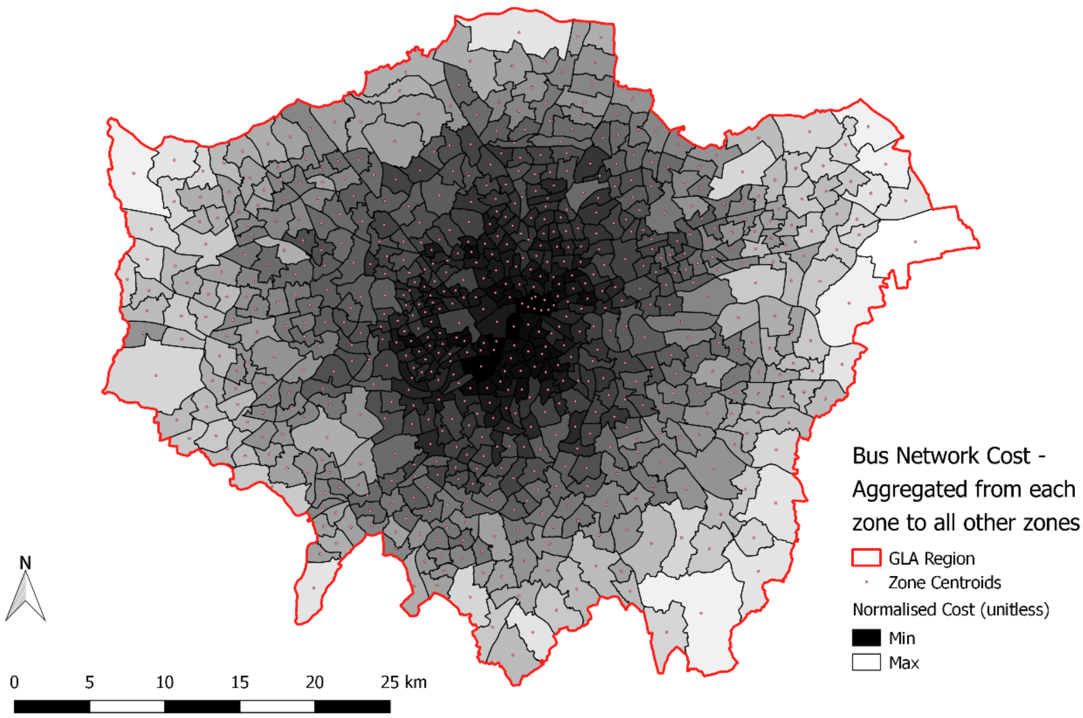

Figure 7.30: Bus Network Generalised Cost.

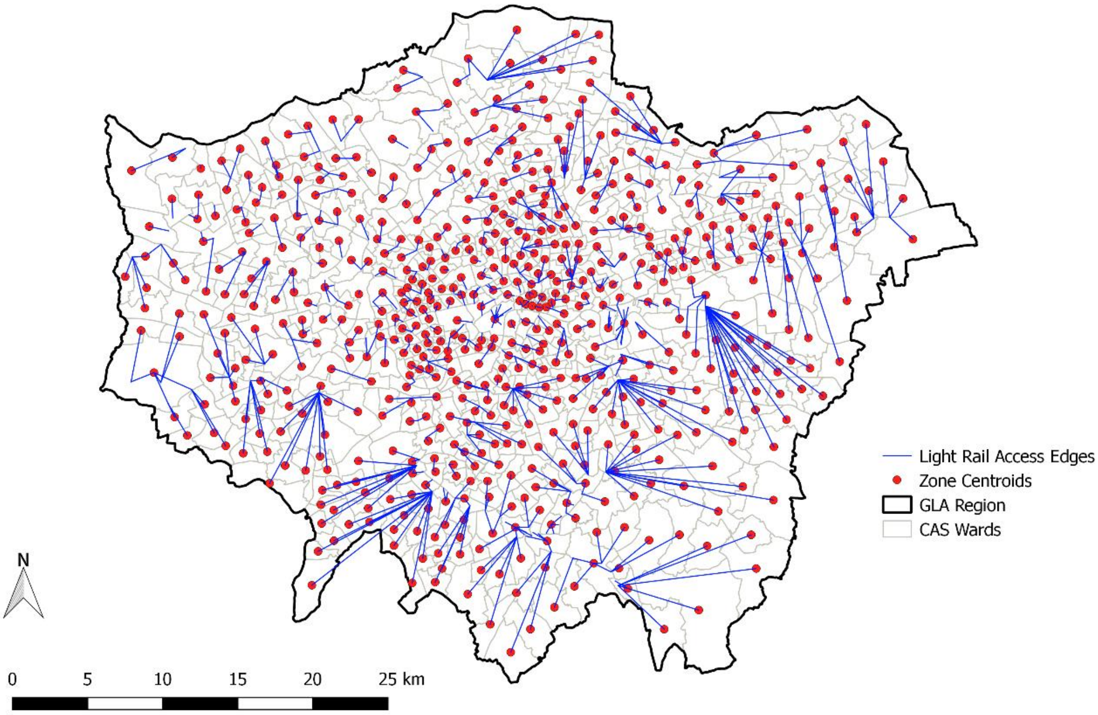

Figure 7.31: Light Rail Network Access Edges.

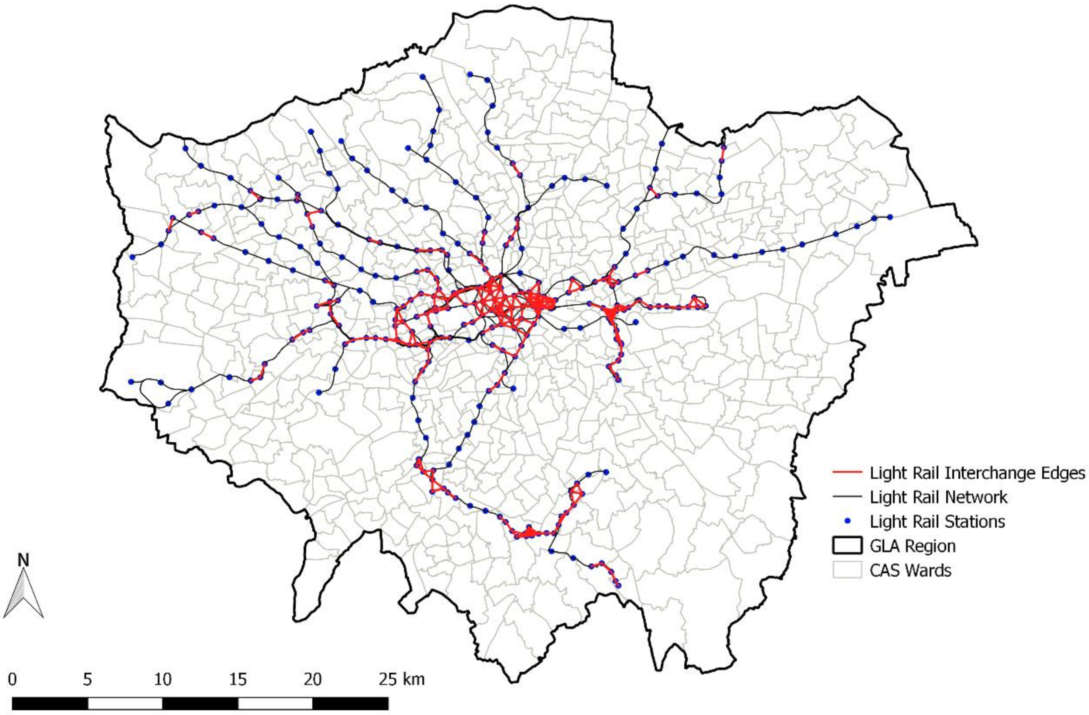

Figure 7.32: Light Rail Network Interchange Edges.

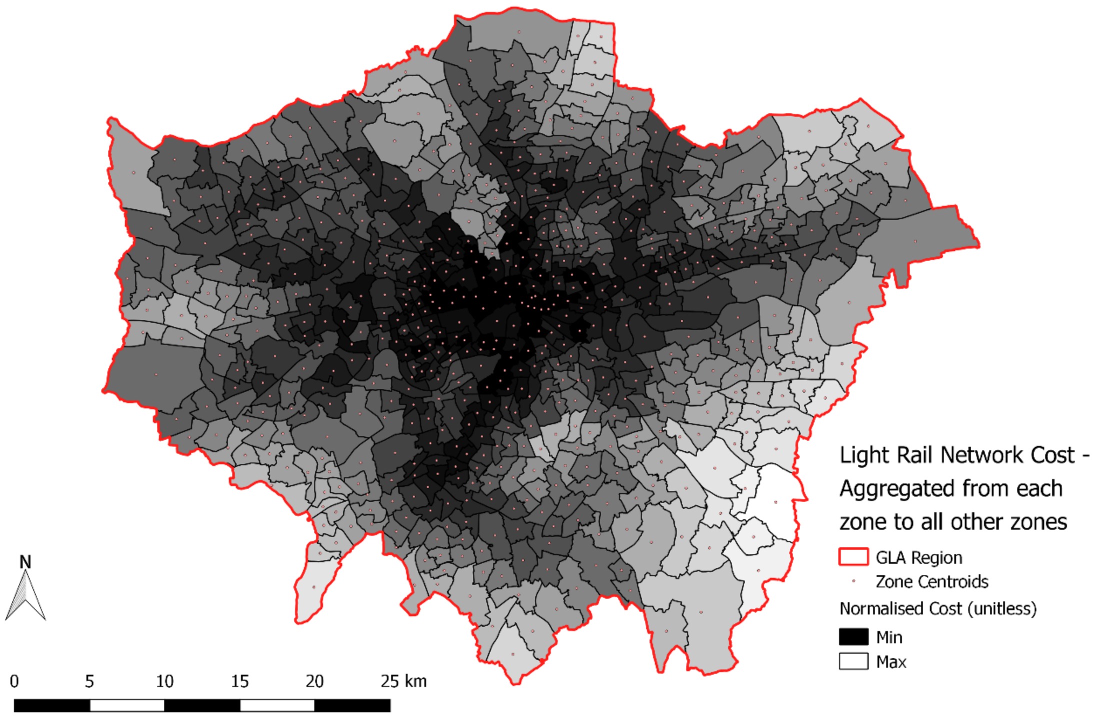

Figure 7.33: Light Rail Network Generalised Cost.

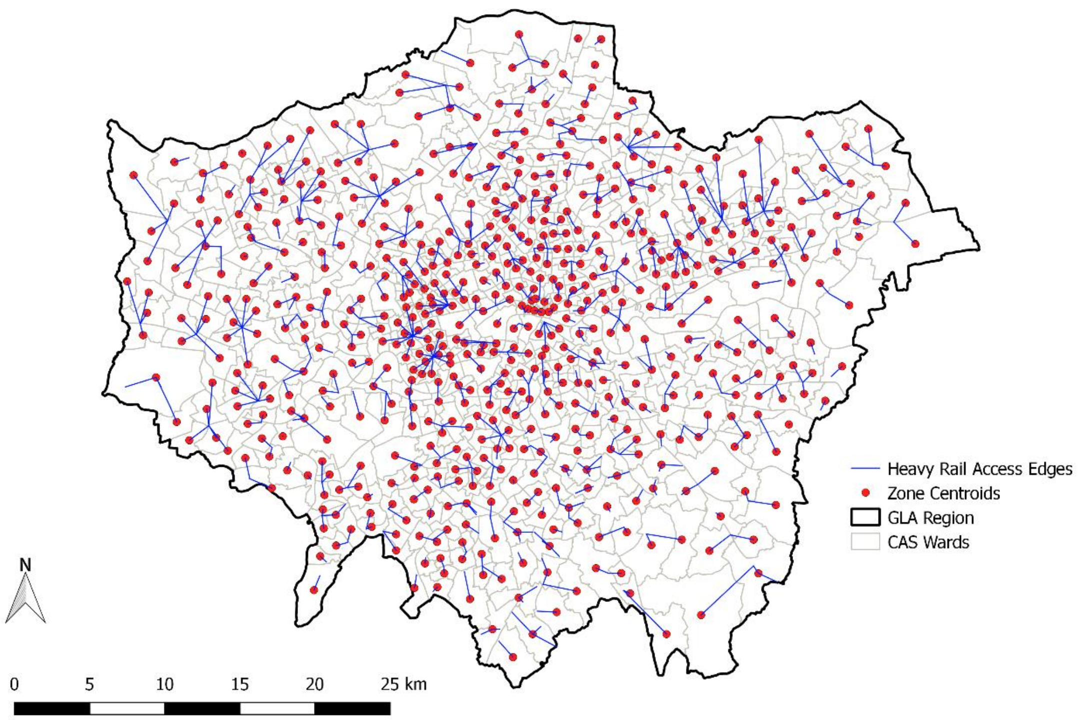

Figure 7.34: Heavy Rail Network Access Edges.

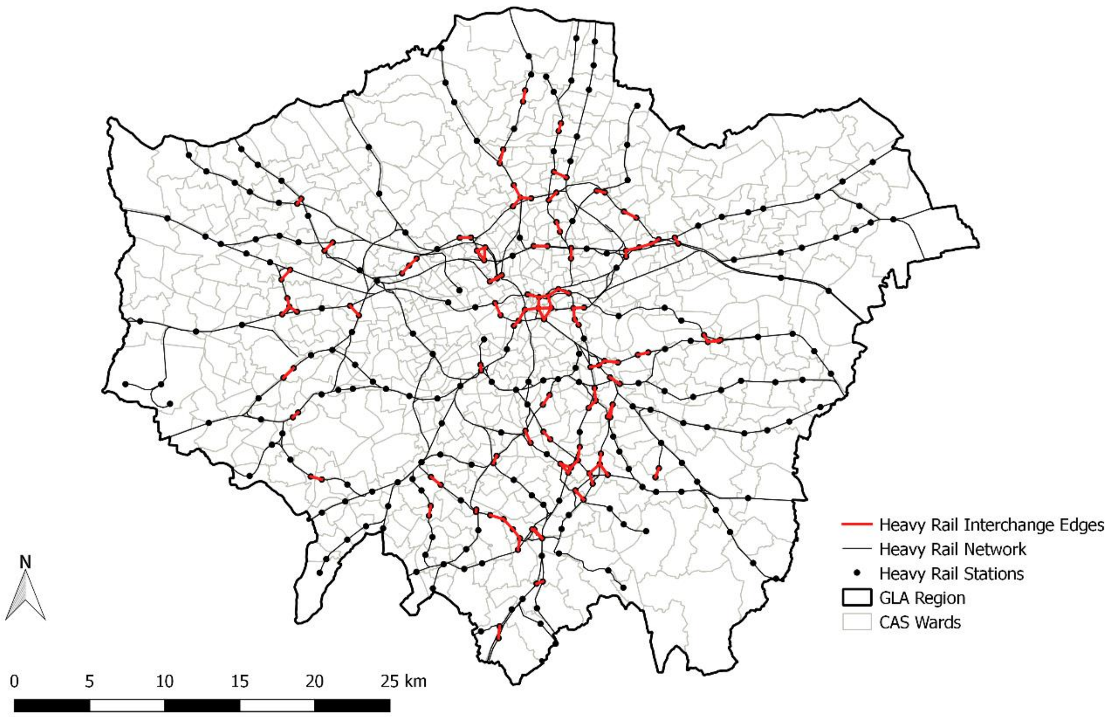

Figure 7.35: Heavy Rail Network Interchange Edges.

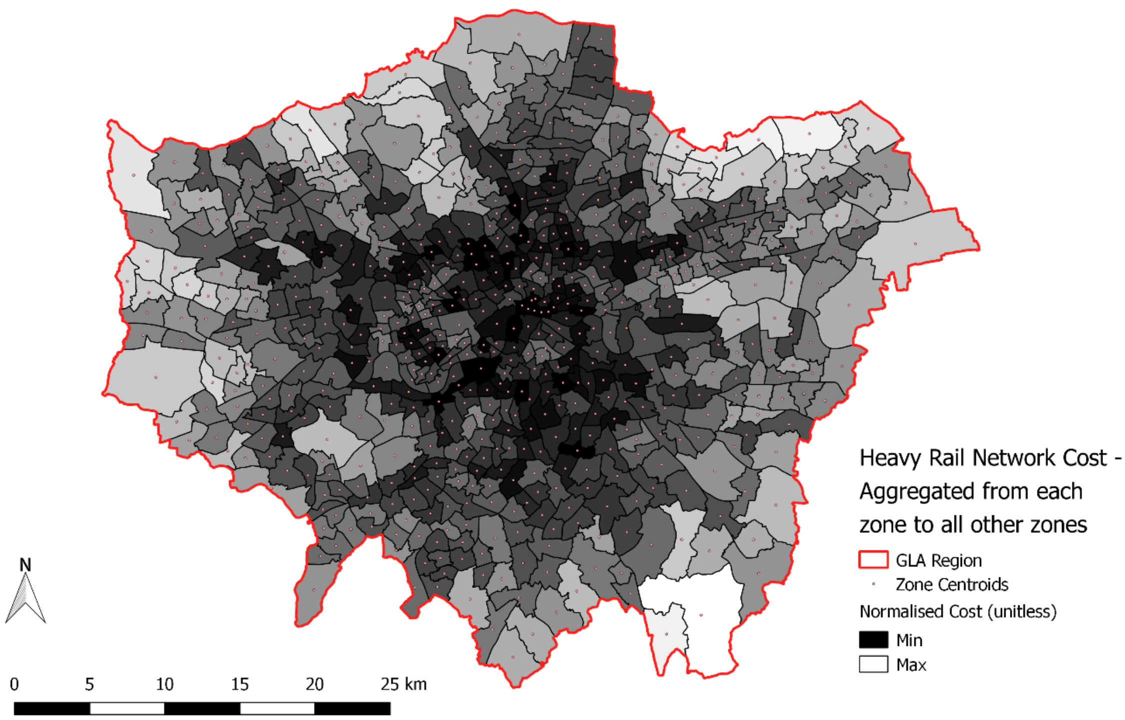

Figure 7.36: Heavy Rail Network Generalised Cost.

The accessibility matrix output from the transport accessibility model (figure 7.37) shows the spread of accessibility values when aggregating costs across all 4 input networks. The radial spread of costs in the road network is evident alongside long access times for modes of public transportation corresponding to zones with poor access to stations and stops.

7.7.2 Spatial Interaction Model

To be consistent with the timescale of the London Plan, figures 7.38 and 7.40 show the zonally disaggregated values of employment and population respectively for the year 2030; these maps provide strong evidence that the spatial attractors described in the London Plan act as the principal drivers of future development. Further extrapolation using these attractors leads to the zonal employment and population maps shown in figures 7.39 and 7.41 respectively, where in the year 2080, the spread of development is roughly the same, allowing for the spread of excess development where targeted zones have reached capacity at the specified density.

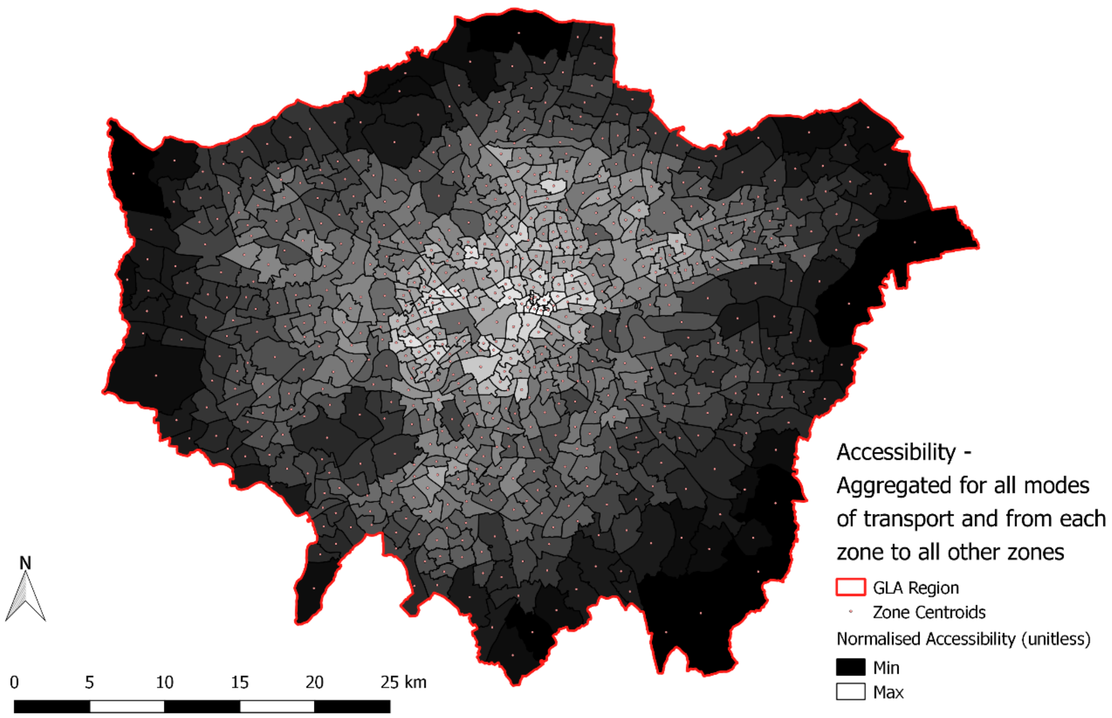

Figure 7.37: Aggregate Transport Accessibility.

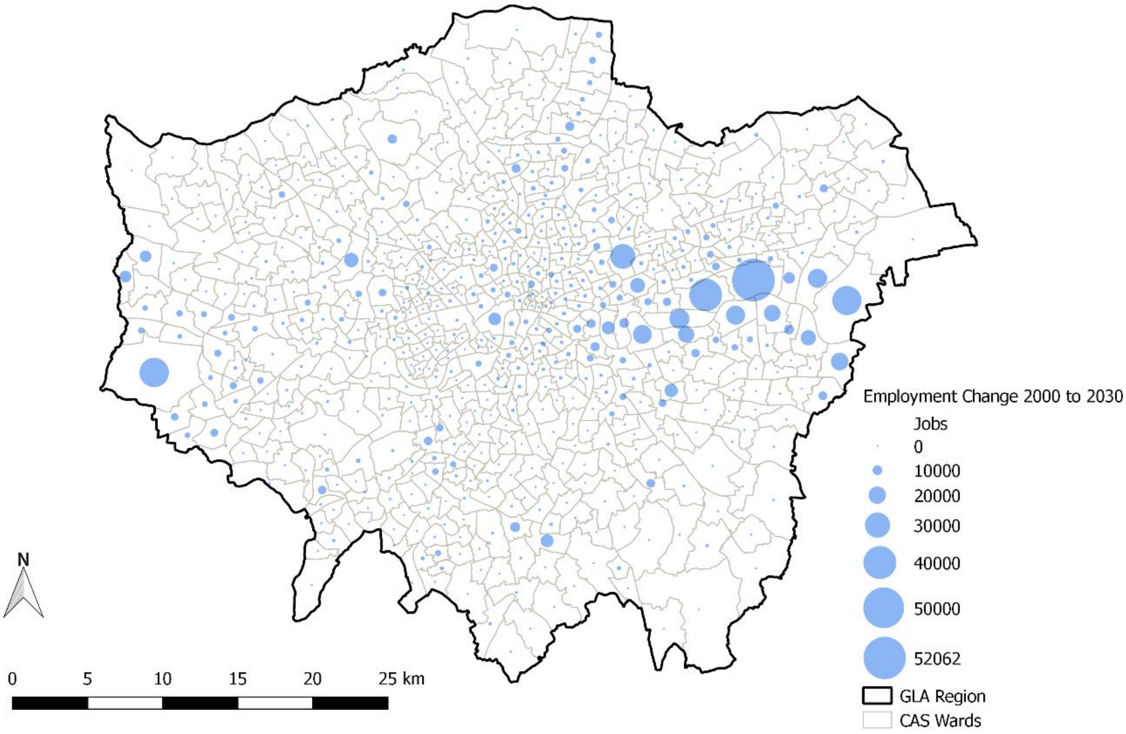

Figure 7.38: 2030 Zonal Employment.

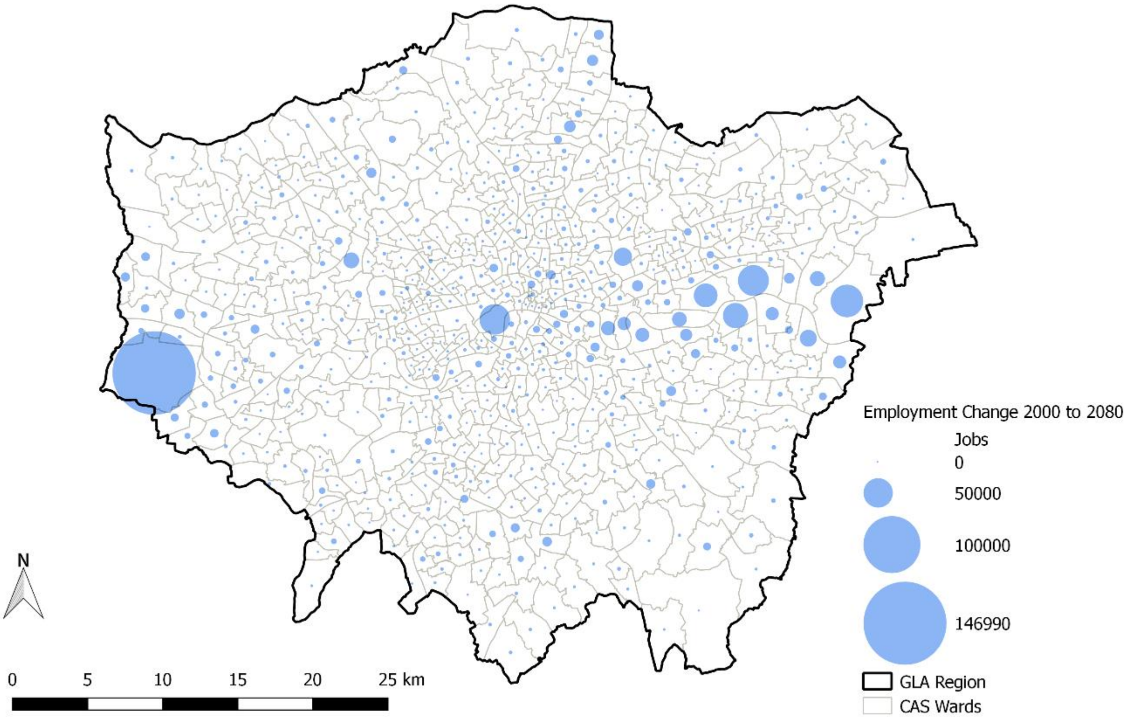

Figure 7.39: 2080 Zonal Employment.

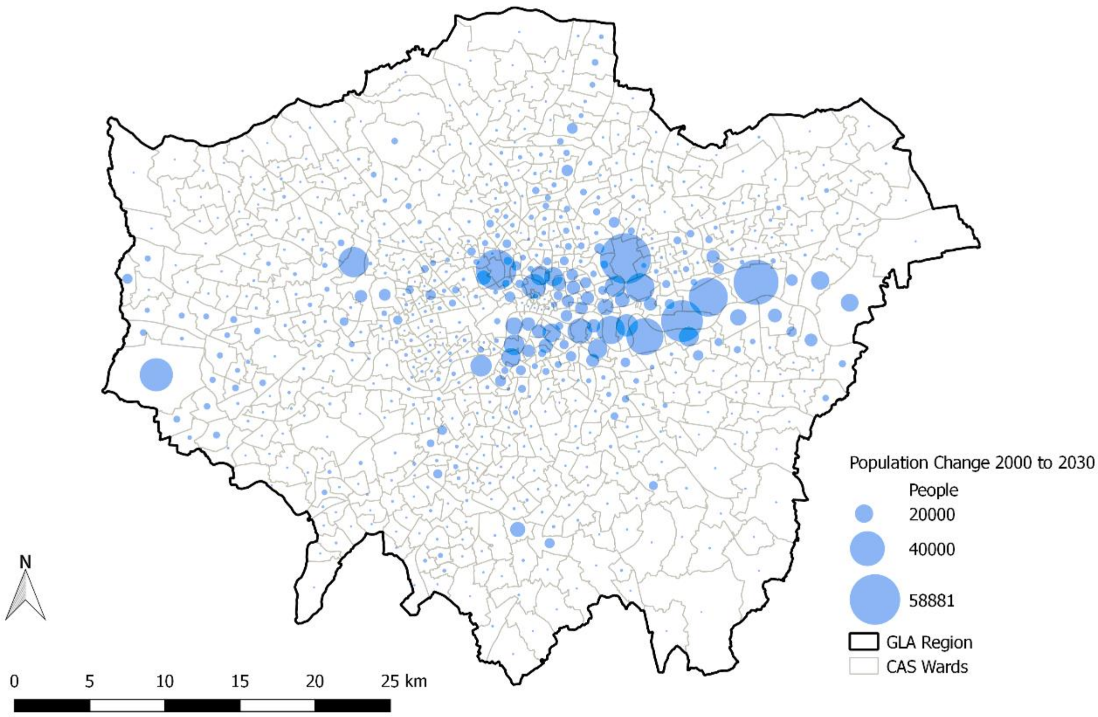

Figure 7.40: 2030 Zonal Population.

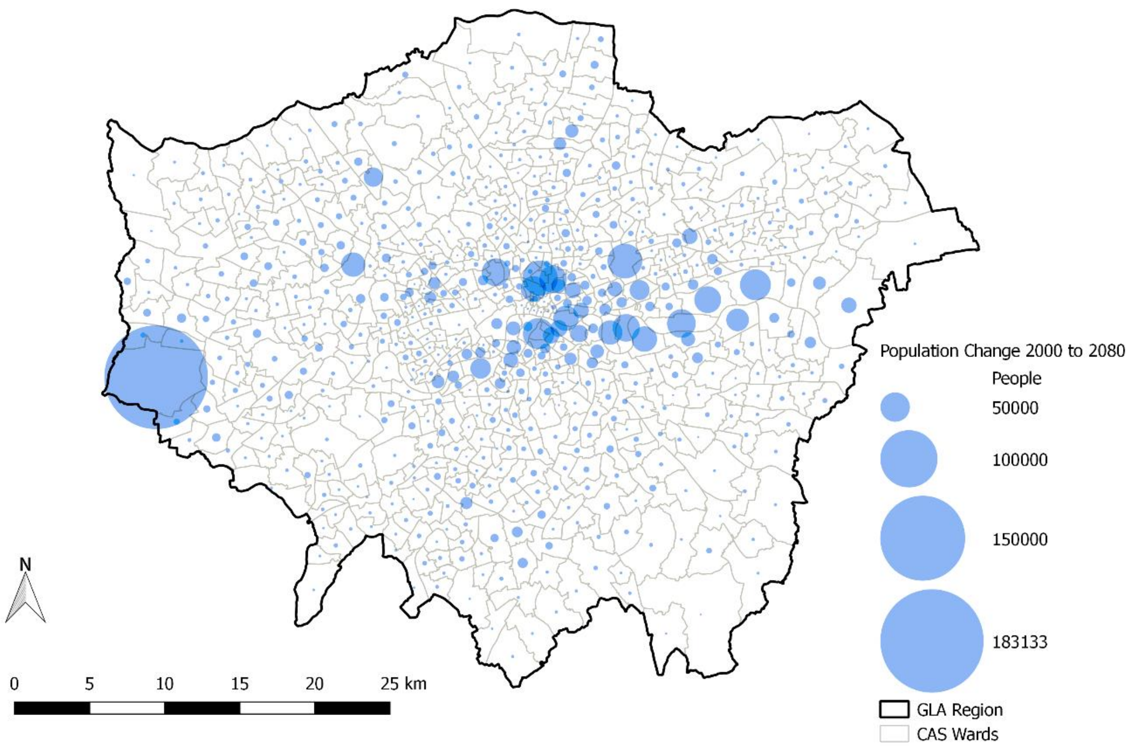

Figure 7.41: 2080 Zonal Population.

7.7.3 Urban Development Model

The first output of the UDM is a raster of development area identity where development area identity values increment from the top-left to the bottom-right of the raster. The minimum zone size parameter, specified as the number of cells is set to 4, which produces the raster shown in figure 7.42, showing the progression of identity values and cells excluded from development areas (shaded white). The second output of the urban development model is a raster of development suitability which is computed using the multi-criteria evaluation of input raster attractors described in section 7.5, and the constraint raster described in section 7.5. Cell development can only occur within valid development areas so a raster of average development suitability (figure 7.44) is output which combines development area identity with the output of multi-criteria evaluation.

Figure 7.45 shows the cellular population development for 2030, corresponding to the output of the zonal model shown in figure 7.39. It is clear the cellular disaggregation of population is in accordance with the SIM output and its input attractors. Figure 7.46 shows the cellular population development for 2080, corresponding to the output of the SIM shown in figure 7.41; this cellular spread of population also correlates with the SIM output and its attractors.

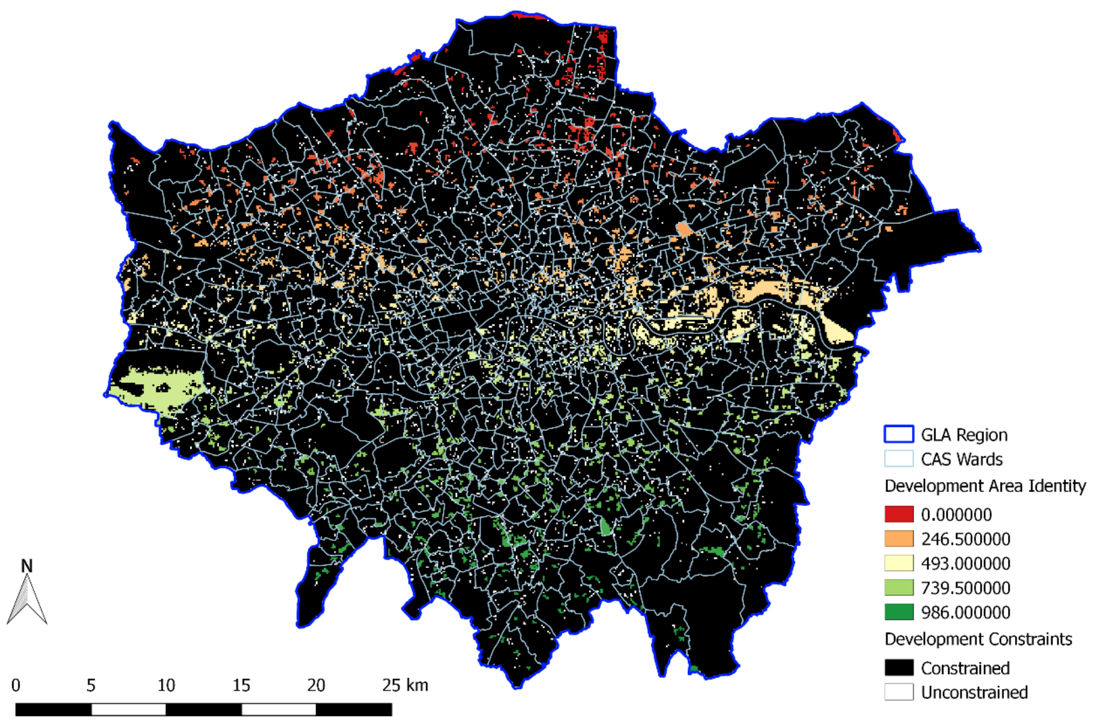

Figure 7.42: Development Area Identity.

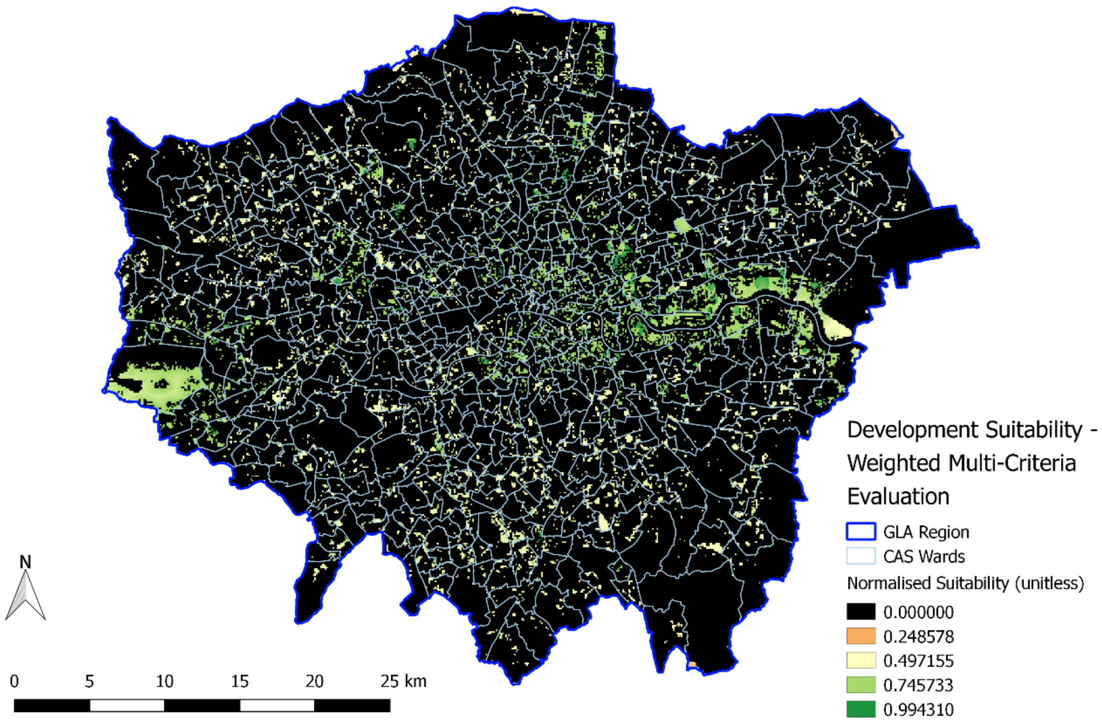

Figure 7.43: Multi-Criteria Evaluation Development Suitability.

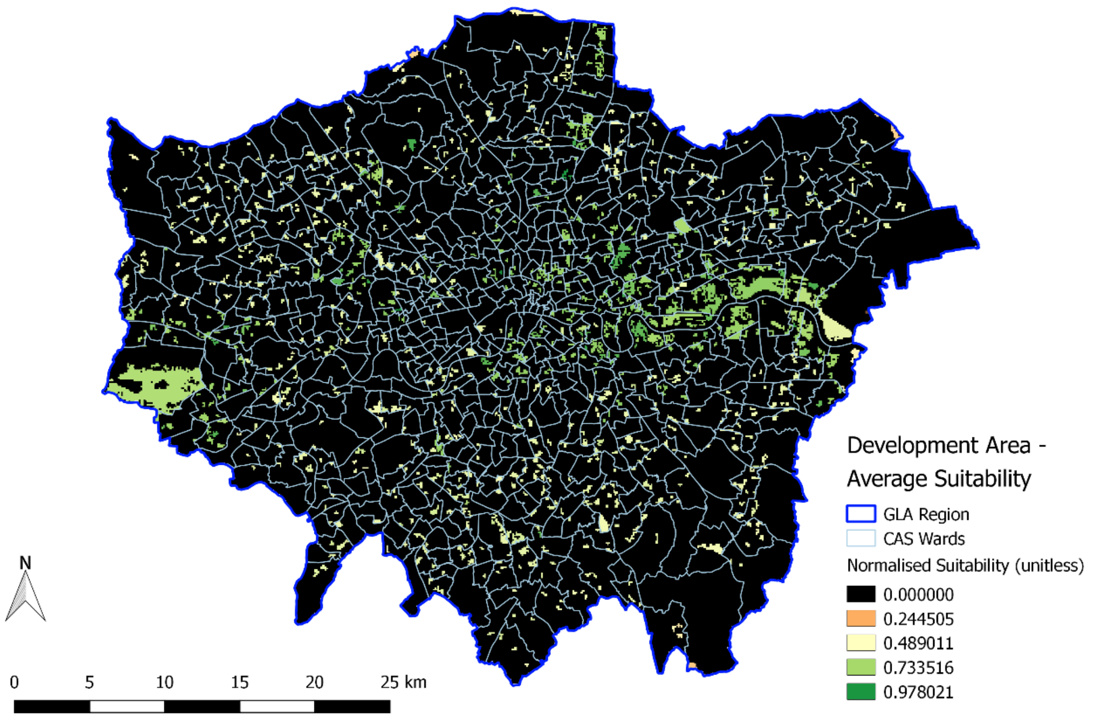

Figure 7.44: Development Area Average Development Suitability.

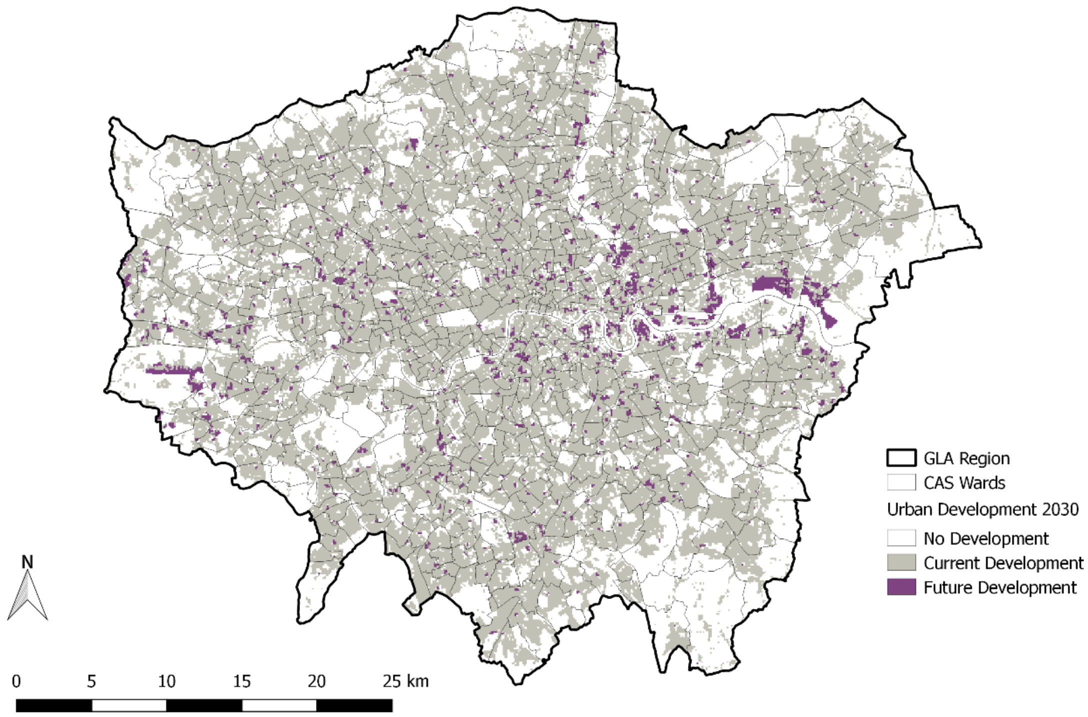

Figure 7.45: 2030 Cellular Population Land Use Development.

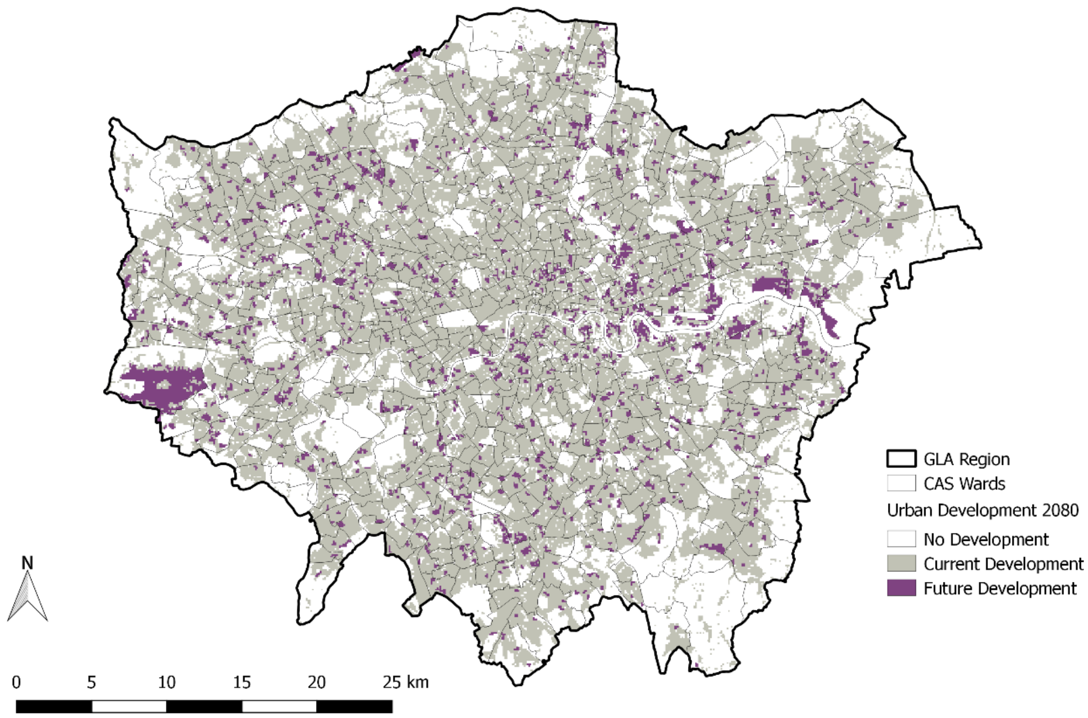

Figure 7.46: 2080 Cellular Population Land Use Development.

7.8 Summary

This chapter has described the application of the UIMF to a case study focussed on the GLA region. A set of detailed spatial policies set out in the Key Diagram of the London Plan was identified and used to form attractors and constraints to the SIM; this was parameterised using observed data and driven using exogenously generated multi-sectoral employment projections from the MDM-E3 model aggregated to five industrial sectors.

Input networks and key parameters used for model parameterisation have been described for each of the four modes of transport in the GLA region. These input networks were modified using the network processing models to include edges for access, interchange and congestion charging. The generalised cost matrix produced for each mode of transport was input to the accessibility model to produce an accessibility matrix aggregated over all transport modes which is input to the SIM.

SIM outputs for employment and population were presented for the year 2030 to match the approximate timeframe of the policies set out in the London Plan. Further outputs were presented for the year 2080 to indicate the possible development resulting from extending the application of these policies far into the future. In all cases, the outputs demonstrate the correct operation of the model in disaggregating future development in accordance with spatial attractors.

The UDM was parameterised with raster data to drive and constrain intra-zonal cell development in a consistent manner with the inter-zonal spatial disaggregation of the SIM. UDM raster outputs were described for development area identity, suitability based on input rasters and obtained using multi-criteria evaluation, and development area average suitability. The final outputs for UDM were cell development rasters for the years 2030 and 2080 which correspond to the SIM outputs presented earlier; the development pattern demonstrates the coherence across modelling scales and shows the progression of development over a 50-year timespan using the same set of attractors and constraints.

This chapter has provided evidence that the UIMF can be applied to the GLA region using a detailed set of spatial policies as inputs. The outputs from each modelling stage show the correct operation of the models themselves and the coherence between models and across modelling stages. Model execution times for each stage in this case study are shown in table 7.6.

Model	Execution time (seconds)
TAM: bus network processing	297
TAM: heavy rail network processing	51
TAM: light rail network processing	38
TAM: road network processing	401
TAM: bus cost and pathfinding	10
TAM: heavy rail cost and pathfinding	4
TAM: light rail cost and pathfinding	4
TAM: road cost and pathfinding	23
TAM: accessibility	4
SIM	2
UDM	4

Table 7.7: GLA Case Study Model Execution Times

The following chapter presents another case study, this time applying the UIMF to Innsbruck in Austria. The focus will be upon driving the model using a relatively sparse set of input data in contrast with the detailed approach taken in this chapter, whilst demonstrating that the modelling framework can be transferred from one study region to another.

Chapter 8. Innsbruck Case Study

8.1 Introduction

This chapter applies the Urban Integrated Modelling Framework (UIMF) to a case study focussed on the city of Innsbruck and its surrounding region; the Inn Valley. Innsbruck has around 125,000 inhabitants (Mikovits *et al.*, 2014) and is situated at an elevation of 574m above sea level in the federal state of Tyrol, in western Austria. The state of Tyrol forms part of the central Alpine ridge and is therefore morphologically defined by mountains and valleys with forest accounting for much of its surface area; this restricts the area available for permanent human settlement. The region is subject to the effects of extreme climate events, particularly rainfall and associated flooding which forms the context of this study.

The purpose of this case study is to demonstrate that the UIMF can be applied to a second study region, used for a different purpose, and can be parameterised with a minimal set of inputs. This shows the utility of the framework, providing further evidence of the models in operation and, in this case, demonstrates transferability.

Section 8.2 introduces the need for simple models to explore possible future scenarios in the context of flooding and urban drainage; sections 8.3 to 8.5 cover model parameterisation, section 8.6 describes the process of model and model group setup within the framework, finally, section 8.7 presents the results generated by each modelling stage from transportation to zonal and sub-zonal land use disaggregation.

8.2 Case Study Context

Land use change resulting from growth and development places further demand on existing critical infrastructure including urban drainage systems. Surface flooding is impacted by the increase in impervious area due to surface sealing for new development, while the additional burden placed on existing drainage can propagate flooding across the system. Development of measures for infrastructure adaptation requires simulation of land use change for a range of scenarios to plan appropriate resilience. Many existing approaches to urban simulation place great demands on resources to parameterise and run models which may be complex, requiring exhaustive inputs and significant computational power. In the context of urban drainage, simpler models have been shown to provide utility in the generation of plausible future scenarios which can then be integrated with models of flooding and drainage infrastructure (Mikovits *et al.*, 2015).

Although the UIMF has relatively modest data requirements as demonstrated for the London case study in Chapter 7, the goal for this case study was to produce a 'working' simulation which was parameterised quickly and made use of as few inputs as possible. This demonstrates that the UIMF has utility in situations characterised by a lack of either time or available data. The key model simplifications are as follows:

- The Spatial Interaction Model (SIM) is driven by linear projections of employment and population for three scenarios.
- Zonal attractors for employment and population use areas designated for future development by regional planning, as well as observed values.
- The Transport Accessibility Model (TAM) uses a single network which carries three modes of transport.
- Generalised cost is simplified to network distance.

Raw input datasets for this case study were provided by Christian Mikovits during a week-long process of guided data-collection which took place at Newcastle University. Census counts were obtained from Statistik Austria for the state of Tyrol, whilst spatial inputs were extracted from open data layers such as OpenStreetMap (OSM).

8.3 Spatial Interaction Model Parameterisation

The study region comprises 140 zones in the state of Tyrol, most of which are situated along the Inn Valley which is centred upon the city of Innsbruck. Figure 8.1 shows this study region which is clipped by a low-resolution raster of habitable area for Austria resulting in a rather un-aesthetic map perimeter.

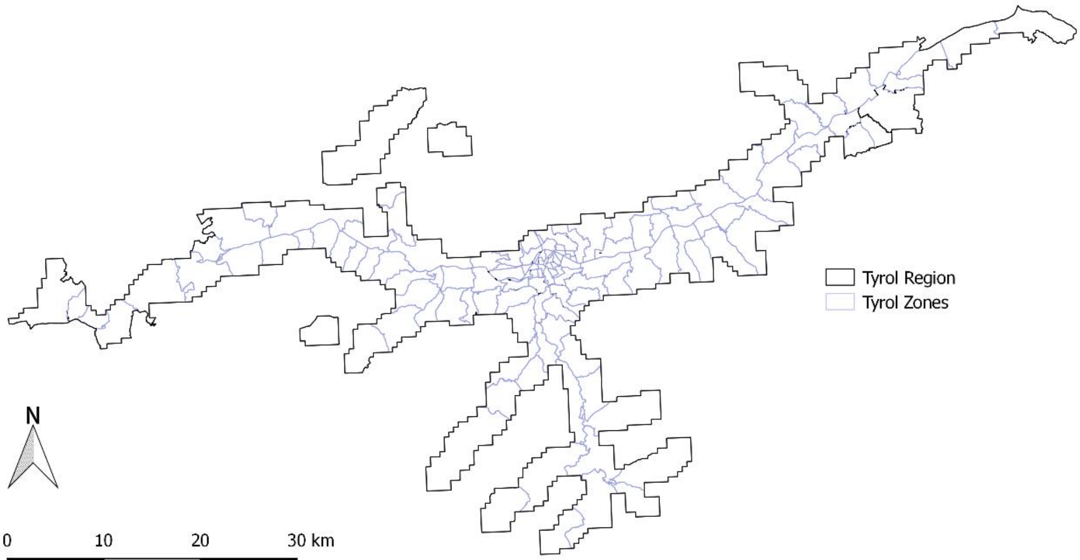

Figure 8.1: Study region comprising 140 zones within Tyrol, Austria.

This case study uses seven timesteps starting from observed values in 2000 and culminates in projections of employment and population for 2030. It makes use of simple linear drivers for three scenarios of projected employment and population: in scenario A, both employment and population increase by 12%; in scenario B, both employment and population increase by 8%; whilst, in scenario C, population is increased by 12% and employment is increased by 8%. These scenarios which are depicted in figure 8.2, demonstrate that simple ratios or percentages can be used to generate offsets to observed values, and drive the model.

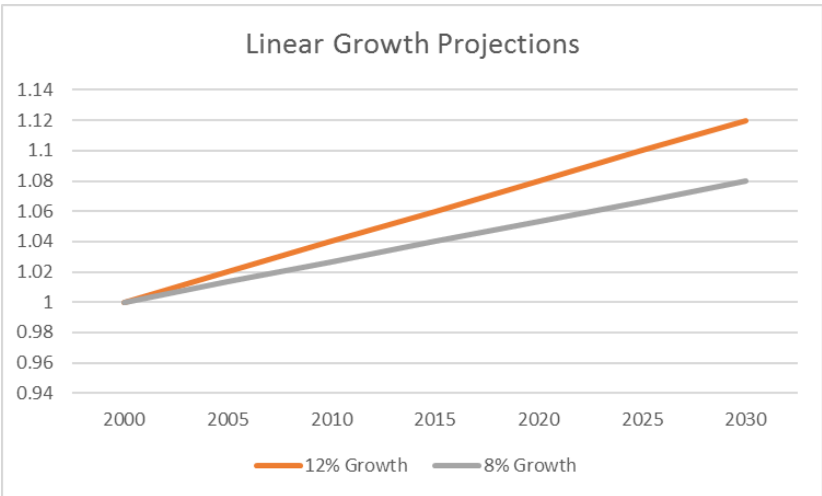

Figure 8.2: Linear growth projections for study region.

As well as observed values which act as agglomerative attractors, the case study employs attractor areas which are consistent with Austrian spatial development planning as shown in table 8.1. *"**Ö**rstliches **R**oum**O**rdnungs**KO**nzept"* roughly translates as "Local Regional Planning Concept" and must be developed by each municipality within Tyrol. The ÖROKO was first developed by the city of Innsbruck in 2002 and identifies future zoning and development planning in a spatial strategy document. A raster of ÖROKO areas designated for structural development including housing and employment was provided as a SIM input as shown in figure 8.3. The function $AreaFromRaster()$ was then used to output a .csv file containing an attractor area value for each zone computed from the number of valid attractor cells multiplied by the cell size squared.

Attractor type	**Attractor 1 (weight)**	**Attractor 2 (weight)**
Employment	Zonal observed employment (1)	Plan area within each zone (1)
Population	Zonal observed population (1)	Plan area within each zone (1)

Table 8.1: SIM Employment and Population Attractors

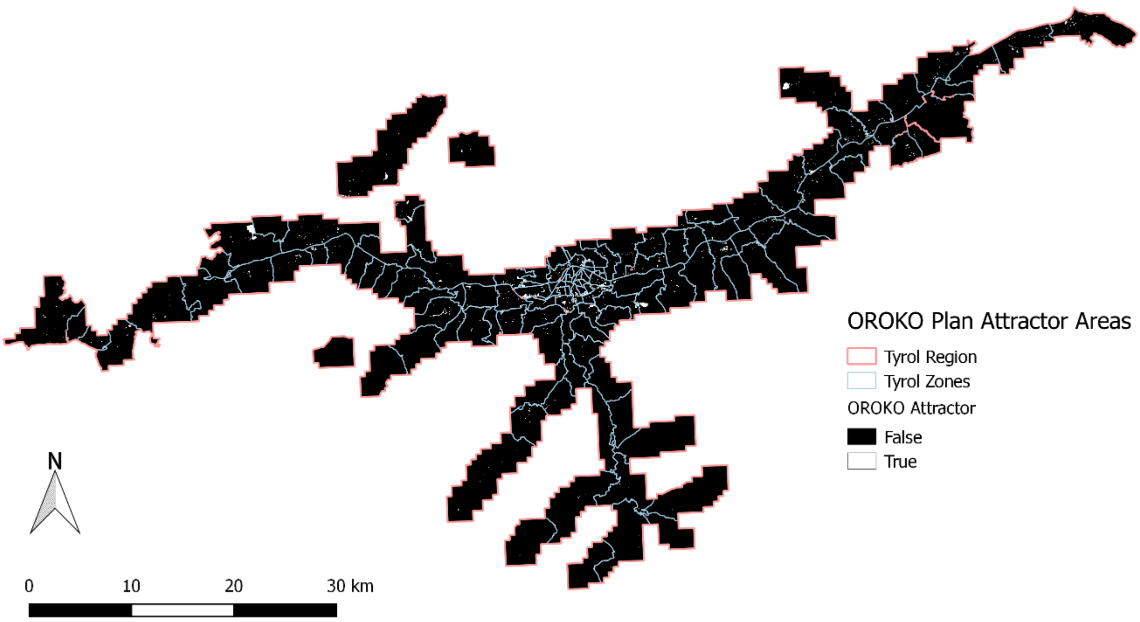

Figure 8.3: ÖROKO designated areas for future development.

A raster of ÖROKO constrained areas, including agriculture, greenspace and landscape protection was provided as a zonal model input as shown in figure 8.4. The function $AreaFromRaster()$ was then used to output a .csv file containing an available area value for each zone computed from the number of free cells multiplied by the cell size squared.

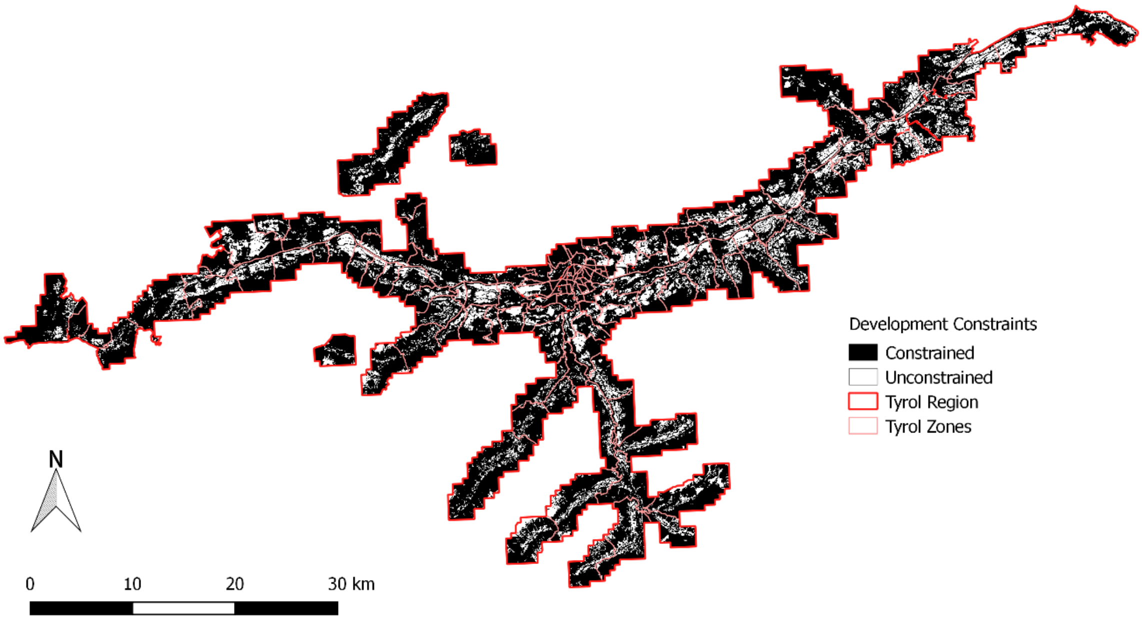

Figure 8.4: ÖROKO constrained areas.

The maps of observed employment (figure 8.5) and employment density (figure 8.6) show that the city of Innsbruck which is divided into numerous relatively small zones, contains a spatially disproportionate number of jobs. Other zones with high employment but relatively low employment density correspond to settlements located along the Inn valley which follows an east-west axis with Innsbruck at the centre. The maps of observed population (figure 8.7) and population density (figure 8.8) show that the city of Innsbruck contains zones with the highest population densities. The settlement pattern along the Inn valley as described for employment is echoed with zones of high population corresponding to, and situated around, zones of high employment. The density values for both employment and population were calculated relative to relevant building footprint area by zone using OSM data.

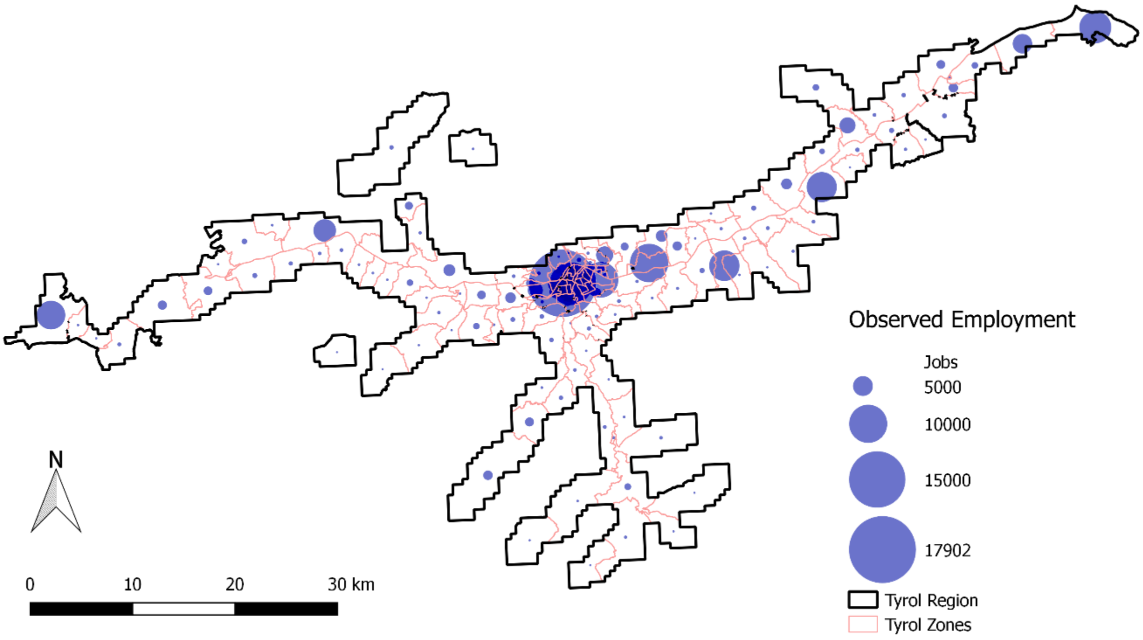

Figure 8.5: Study region observed employment.

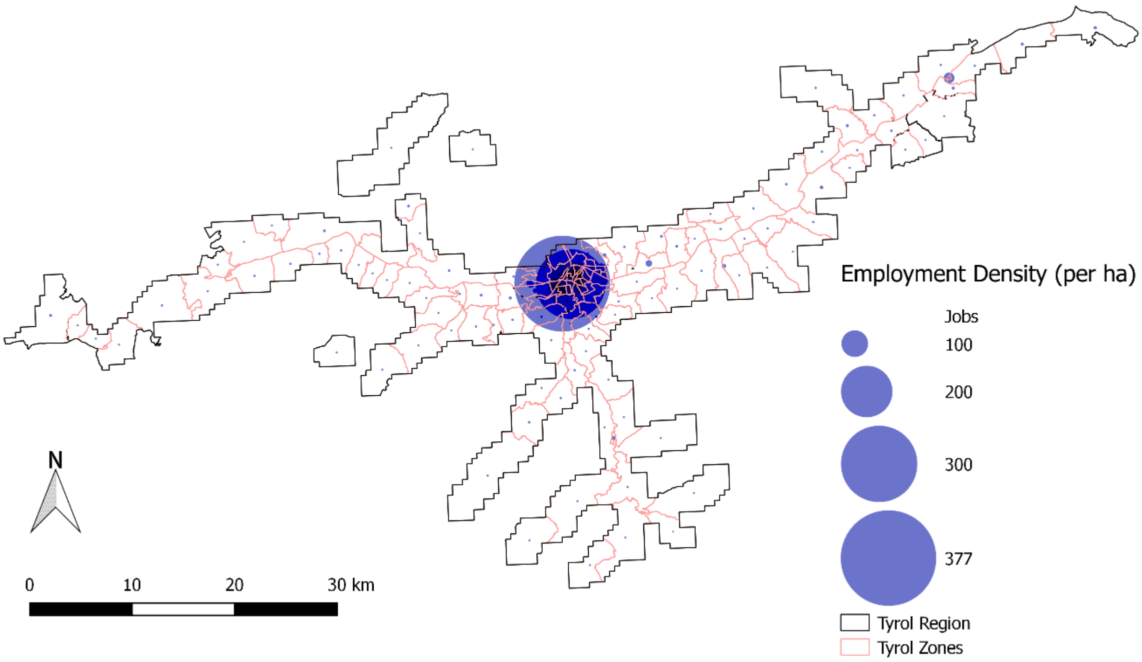

Figure 8.6: Study region employment density.

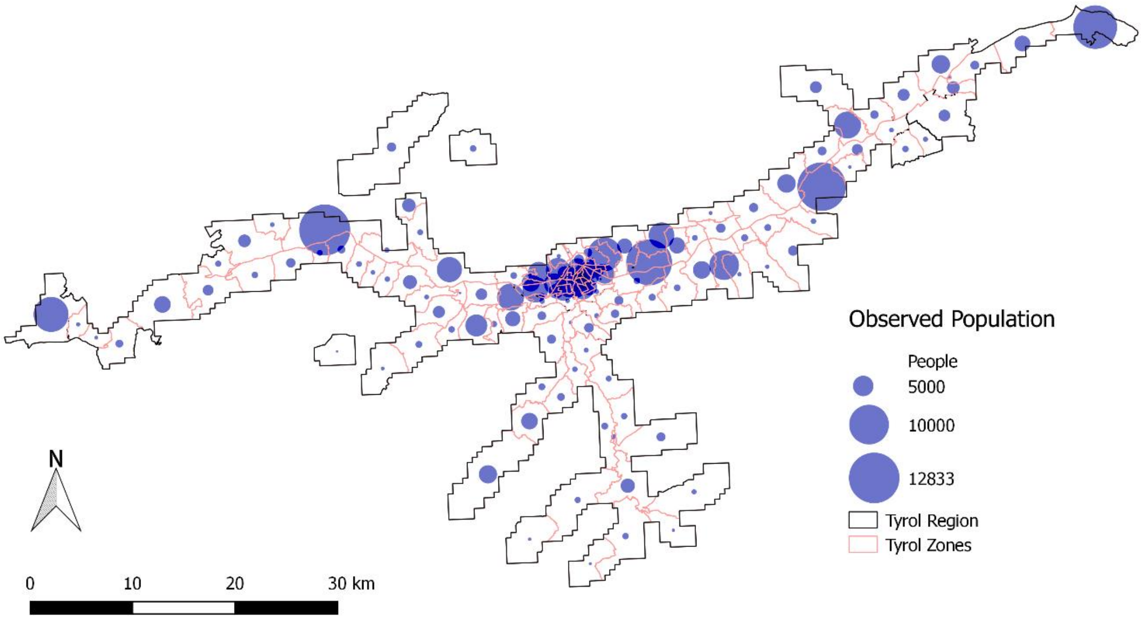

Figure 8.7: Study region observed population.

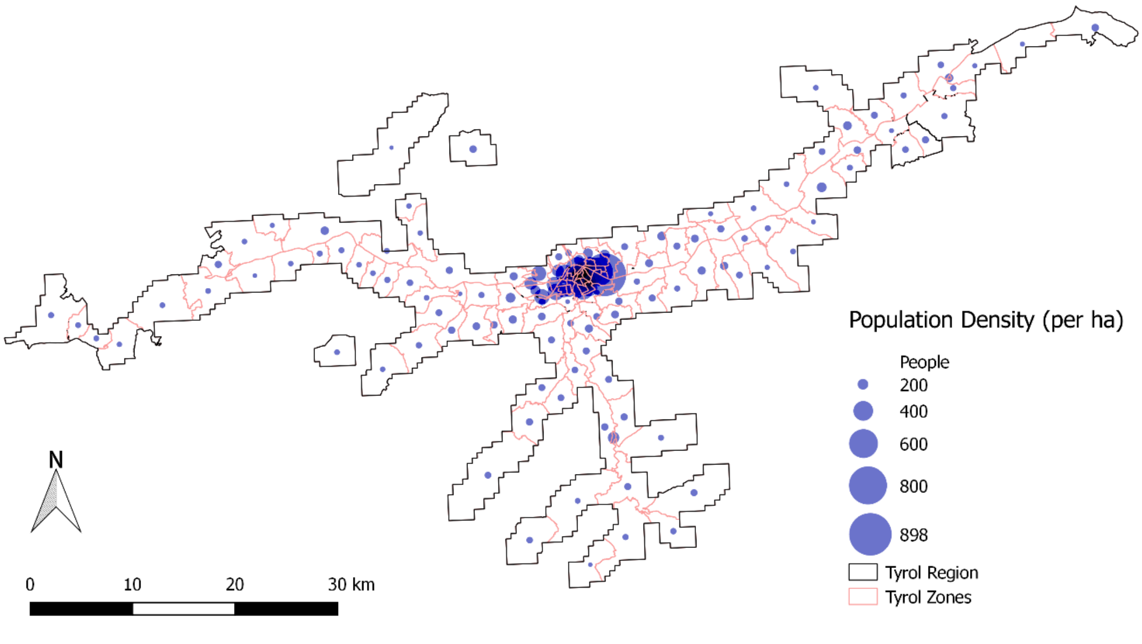

Figure 8.8: Study region population density.

8.4 Transport Accessibility Model Parameterisation

In this case study, for the purposes of simplification, transport accessibility is represented by a single network as shown in figure 8.9. The road network carries three modes of transport; private vehicles, cycles and buses. As a further simplification measure, all network edges are set to the same speed which effectively configures the Dijkstra algorithm to find shortest distance paths between all zones. Early spatial interaction models used the straight-line Euclidean distance between zones as a measure of separation in their gravity models of spatial disaggregation; in practice, this would result in straight-line routes connecting zones through mountains which suggests that network distance, as used in this case study, is a more sensible choice as a basic measure of separation.

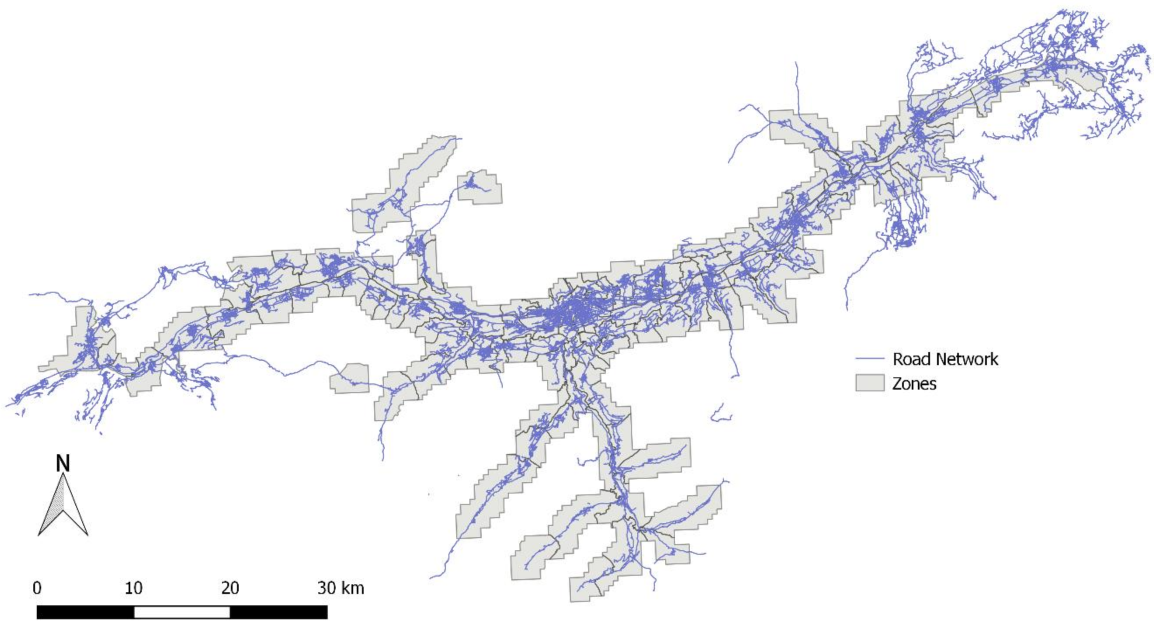

Figure 8.9: Study region input road network.

8.5 Urban Development Model Parameterisation

A raster of integer zone identity values must be created for the Urban Development Model (UDM) to correctly assign cells to zones. These values correspond to the index to the zone array used in the urban development model.

To ensure consistency across modelling scales the ÖROKO attractors used in the SIM (figure 8.3) for inter-zonal population disaggregation are used in the urban development model for intra-zonal population disaggregation at the cellular level as shown in table 8.2. Inter-zonal population disaggregation is also driven by accessibility to employment via the transport model so intra-zonal population disaggregation employs localised accessibility to transportation; the raster of proximity to the road network (shown in figure 8.10) is used for this purpose. The tendency for new development to agglomerate around existing development is also represented at the intra-zonal level by using rasterised proximity to current development as shown in figure 8.11.

Suitability Attractors	Attractor Weight
Plan area	0.4
Development proximity	0.3
Road proximity	0.3

Table 8.2: UDM Suitability Attractors

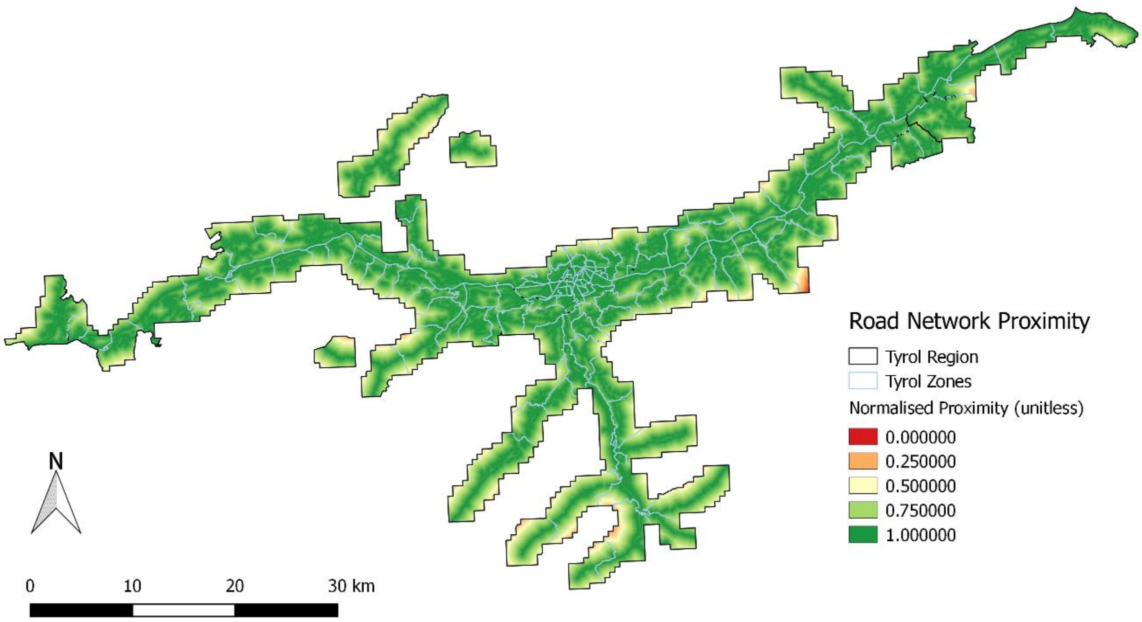

Figure 8.10: Rasterised proximity to road network.

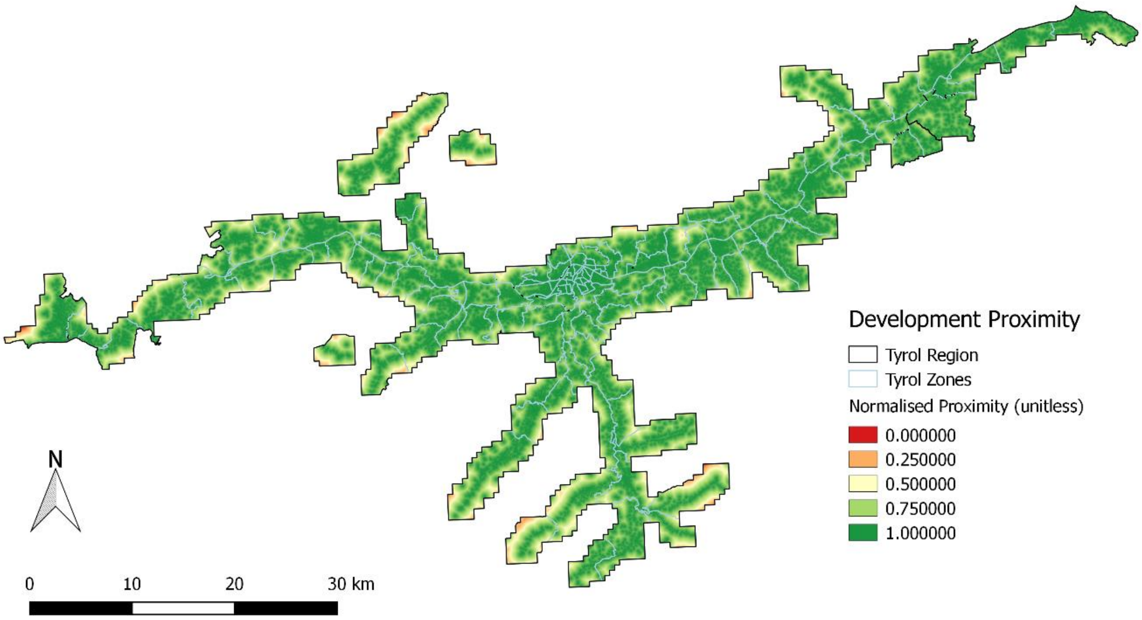

Figure 8.11: Rasterised proximity to current development.

As well as using consistent attractors, coherence across modelling scales is ensured by using consistent constraints which are used to define the available area for new development in both the SIM and UDM. The raster of ÖROKO constrained areas was shown in figure 8.4.

8.6 Framework Setup

A shown in figure 8.12, five models are required to model the Innsbruck region; a network processing model and generalised cost model for the road network, followed by an accessibility model to convert the input cost matrix. The output accessibility matrix is fed into the SIM which, in turn, generates zonal population data to be used by the UDM. As shown in figure 8.13, PostgreSQL databases are used to configure and run the collection of models for the Innsbruck region. Model and model group data in .csv format is prepared in a spreadsheet using a template table for each dataset before loading to the $CSInnsB\_ModelGroups$ and $CSInnsB\_RunData$ databases. Raster data is prepared in $QGIS$ and exported using the ascii file format before converting and loading into the $CSInnsB\_RunData$ database. Vector data is prepared in $QGIS$ and imported to the $CSInnsB\_RawData$ database using the shapefile import/export plugin in $pgadmin$. This vector data is then processed to extract the required columns of data, reformatting where necessary and loading into tables in the $CSInnsB\_RunData$ database.

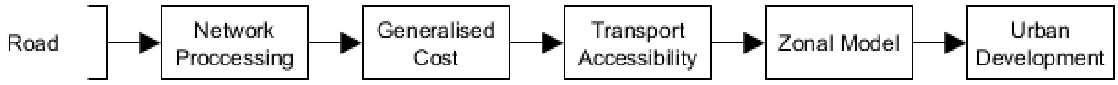

Figure 8.12: Study region models.

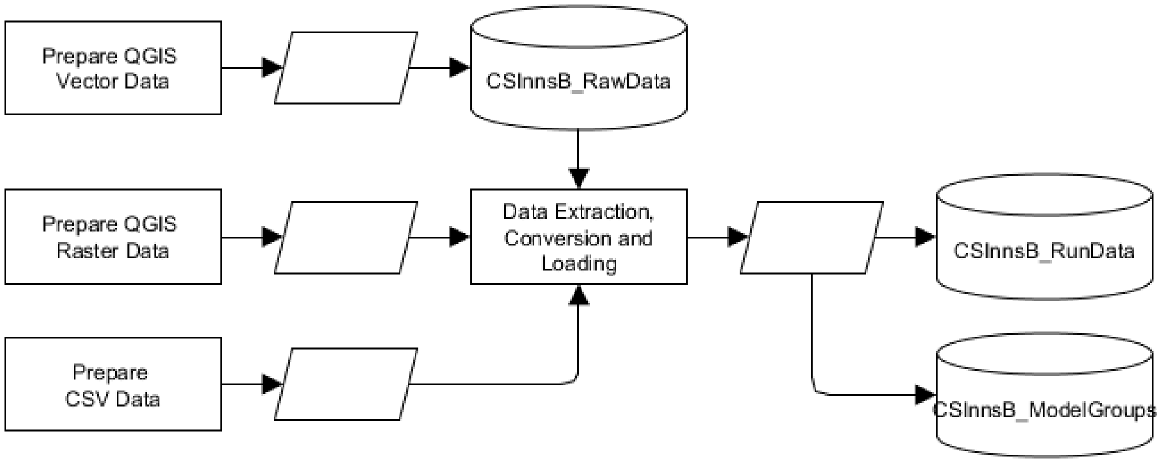

Figure 8.13: Study region model setup.

8.7 Results

8.7.1 Transport Accessibility Model

The distance-based cost matrix for the road network which is shown in figure 8.14 exhibits the expected radial pattern of transport cost centred on the city of Innsbruck. Figure 8.15 shows this pattern converted into accessibility in which a higher cost of travel between zones leads to lower accessibility. A more detailed representation of transport accessibility for the study region would incorporate all available modes of transport and make use of full generalised cost parameterisation; however, this greater detail comes at the expense of increased data requirements.

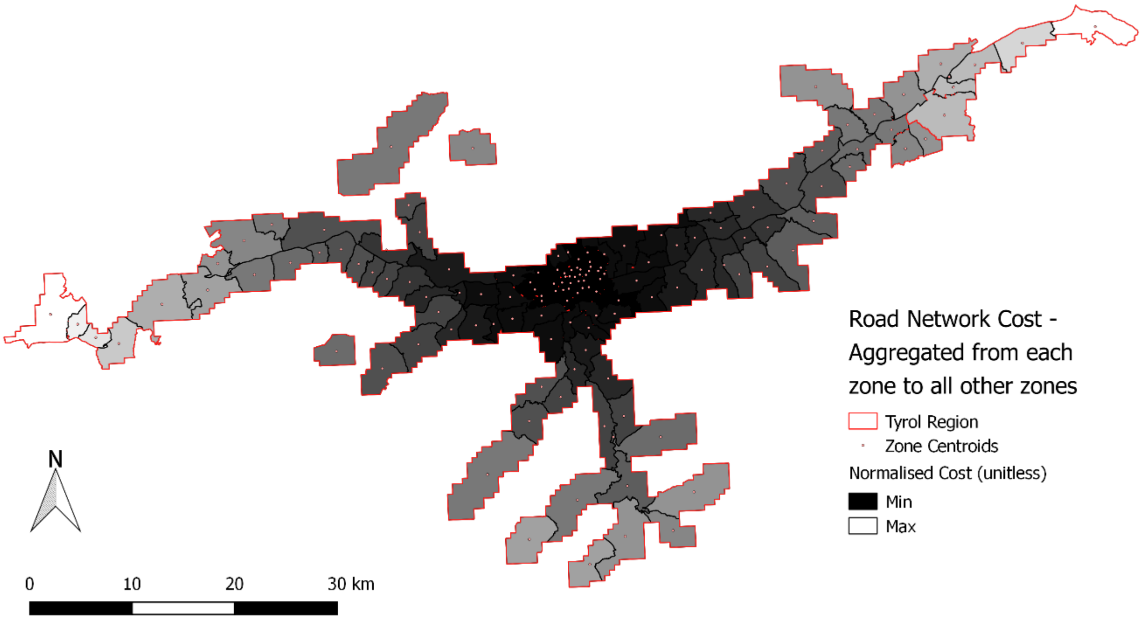

Figure 8.14: Study region network distance-based cost.

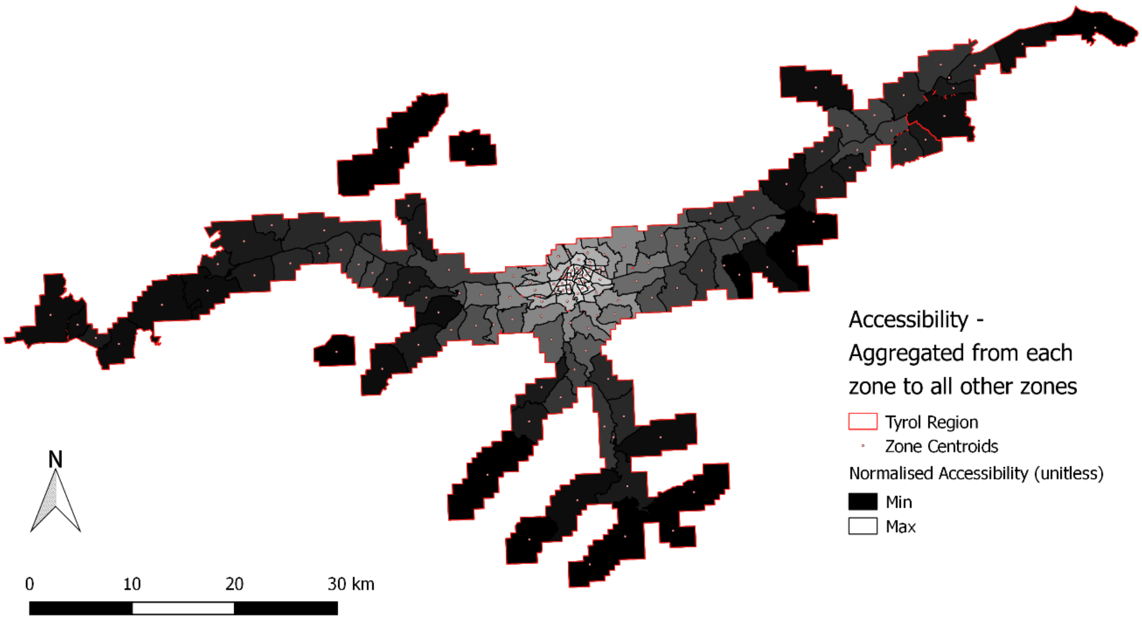

Figure 8.15: Study region employment accessibility.

8.7.2 Spatial Interaction Model

To simplify the task of basic analysis, all maps of zonal outputs are presented as the change in values from 2000 to 2030. The linear projection scenarios for employment and population described in section 8.3 were used to produce three sets of outputs which were then used to drive the UDM. Only scenarios A and C are presented here since scenario B simply reduces scenario A employment and population growth from 12% to 8%, resulting in a similar spatial spread with scaled back values. As shown in figure 8.16, scenario A employment change (12%) reflects the attraction to observed employment (figure 8.5) and designated ÖROKO areas (figure 8.3) subject to available space. Figure 8.17 shows scenario A population change which reflects the attraction to observed population (figure 8.7) and designated ÖROKO areas (figure 8.3); however, population disaggregation is heavily influenced by accessibility to employment (figure 8.15) which leads to most future development occurring close to the city of Innsbruck.

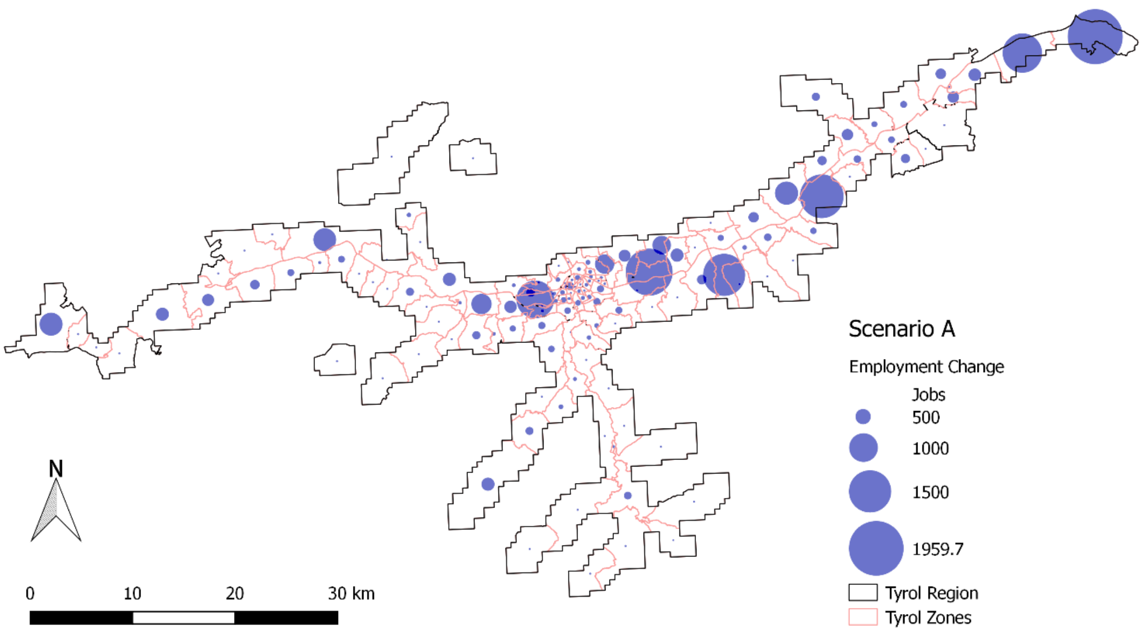

Figure 8.16: Scenario A employment change.

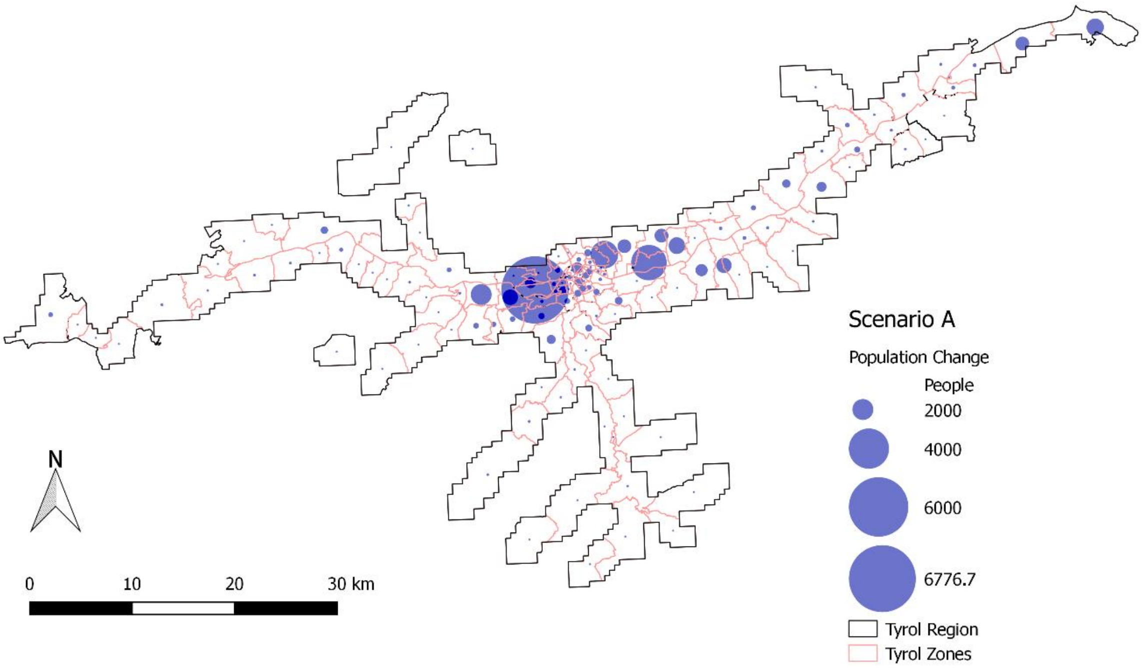

Figure 8.17: Scenario A population change.

As shown in figure 8.18, scenario C employment change (8%) reflects the same spatial attractors as for scenario A (figure 8.16) but is scaled back and must compete for available space with the faster growing population. Figure 8.19 shows scenario C population change (12%) which is spatially similar to scenario A (figure 8.17) but the slower growing economy and associated land use change permits more development for housing in zones with higher accessibility in and around the city of Innsbruck.

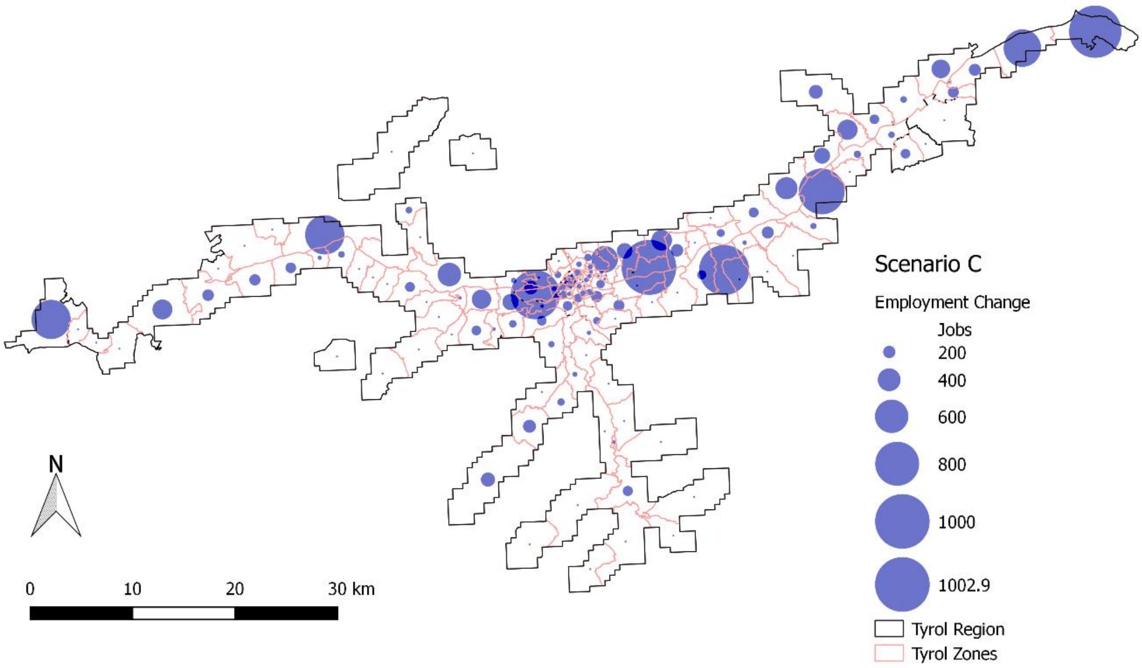

Figure 8.18: Scenario C employment change.

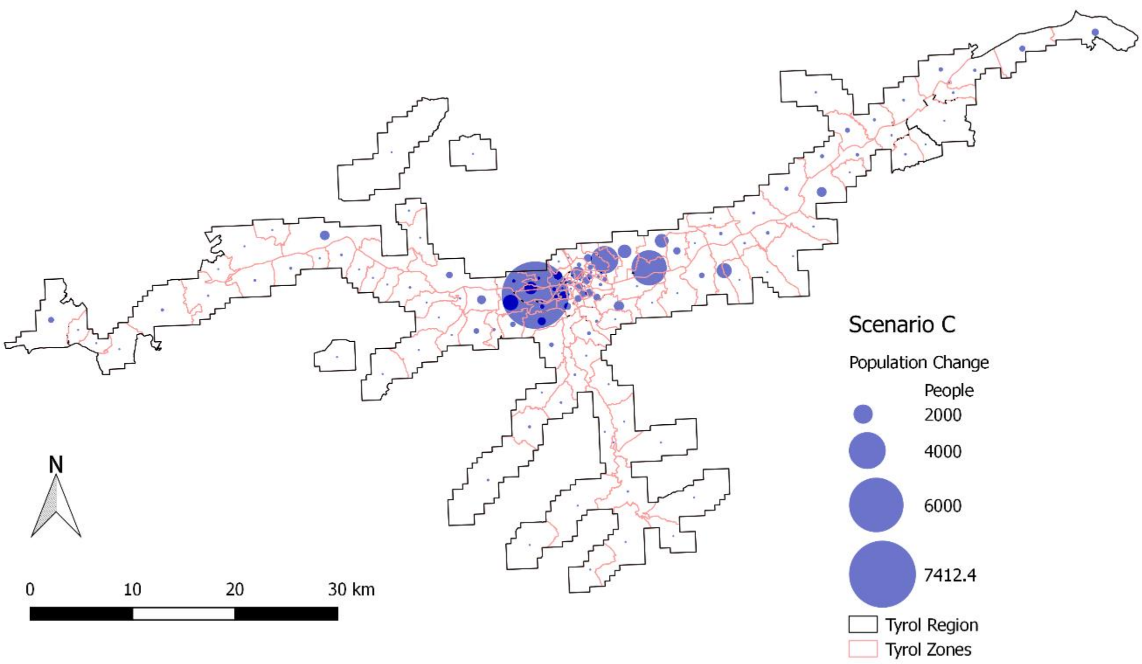

Figure 8.19: Scenario C population change.

Zonal population disaggregation is driven by accessibility to employment which in this case study has been greatly simplified using network distance as a measure of separation; this introduces a strong bias towards the central region around the city of Innsbruck which ignores the various modes of transport available and the associated time and cost of travel. To quickly generate an alternative view of accessibility for comparison, the SIM was driven using a uniform accessibility matrix in which all values are set to one; although this represents an extreme and unlikely situation where network distance is offset by edge speeds and journey costs it is useful to demonstrate the effect of accessibility on population disaggregation. Figure 8.20 shows scenario C population change (12%) driven by a uniform matrix, by nullifying accessibility the change in population is spread throughout the study region reflecting the attraction to observed population (figure 8.7) and designated ÖROKO areas (figure 8.3).

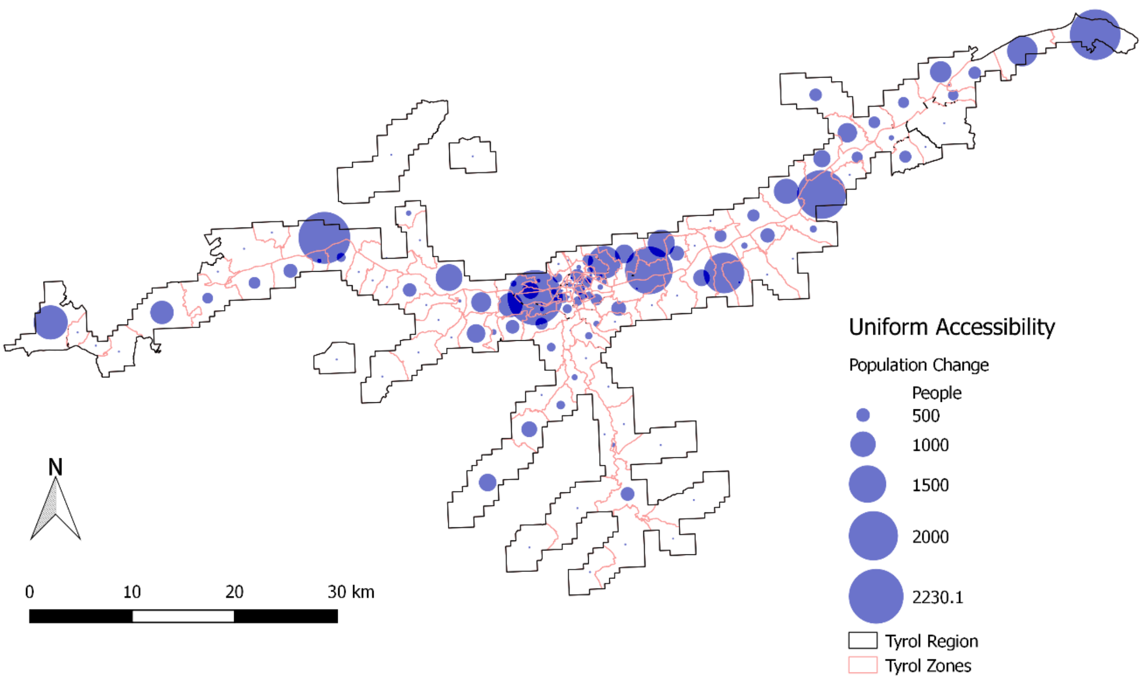

Figure 8.20: Scenario C uniform accessibility population change.

8.7.3 Urban Development Model

The first output of the UDM is a raster of development area identity where development area identity values increment from the top-left to the bottom-right of the raster. The minimum zone size parameter, specified as the number of cells is set to1, which produces the raster shown in figure 8.21, showing the progression of identity values. The second output is a raster of development suitability which is computed using the multi-criteria evaluation of input raster attractors described in section 8.5, and the constraint raster described in section 8.5.

Cell development can only occur within valid development areas so a raster of average development suitability (figure 8.22) is output which combines development area identity with the output of multi-criteria evaluation.

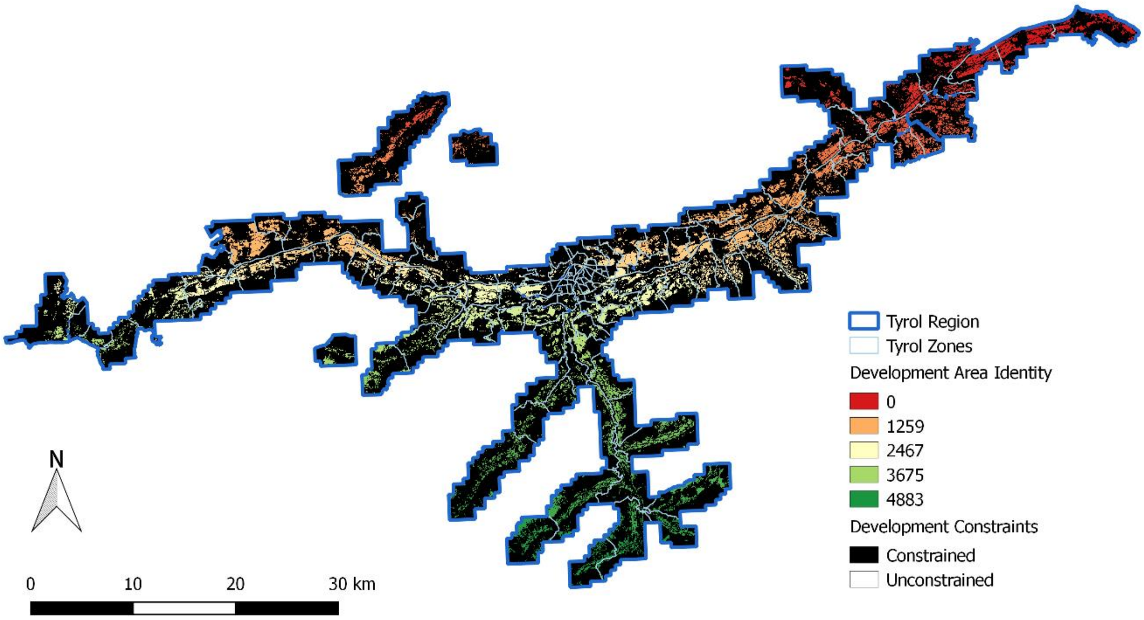

Figure 8.21: Study region development area identity.

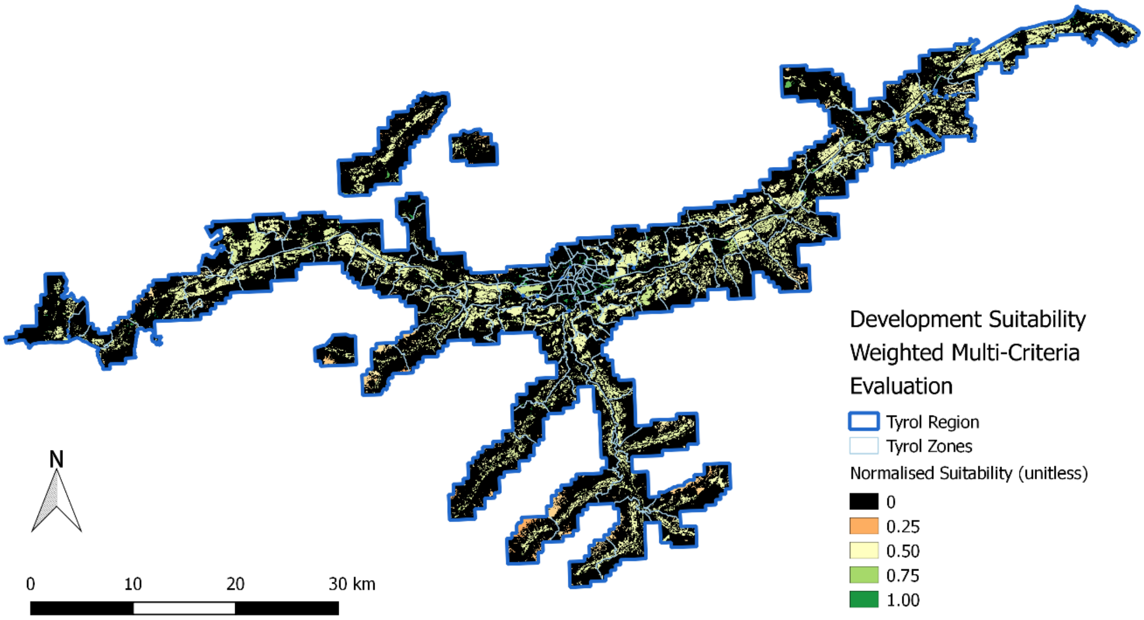

Figure 8.22: Study region development area average suitability.

Figure 8.23 shows the cellular population development for 2030, corresponding to the output of the SIM for scenario A shown in figure 8.17. The cellular disaggregation of population is in accordance with the SIM output and its input attractors. Figure 8.24 shows the cellular population development for 2030, corresponding to the output of the SIM for scenario C shown in figure 8.19. The land use change associated with slower economic growth permits more development for housing in zones in and around the city of Innsbruck which is reflected in the pattern of cellular development. Figure 8.25 shows the cellular population development for 2030, corresponding to the output of the SIM for scenario C using uniform accessibility shown in figure 8.20. The cellular development pattern corresponds to the spread of population throughout the study region reflecting the attraction to observed population (figure 8.7) and designated ÖROKO areas (figure 8.3).

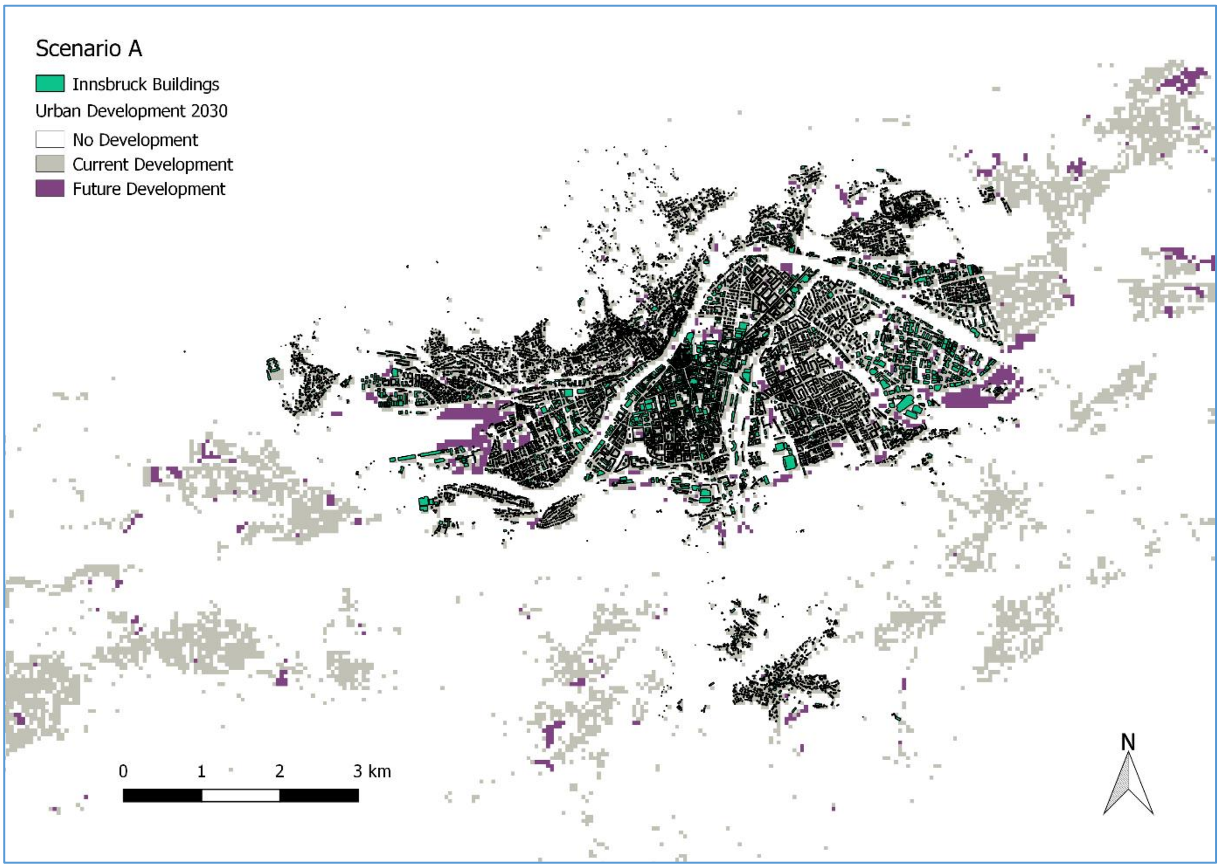

Figure 8.23: Scenario A cellular development.

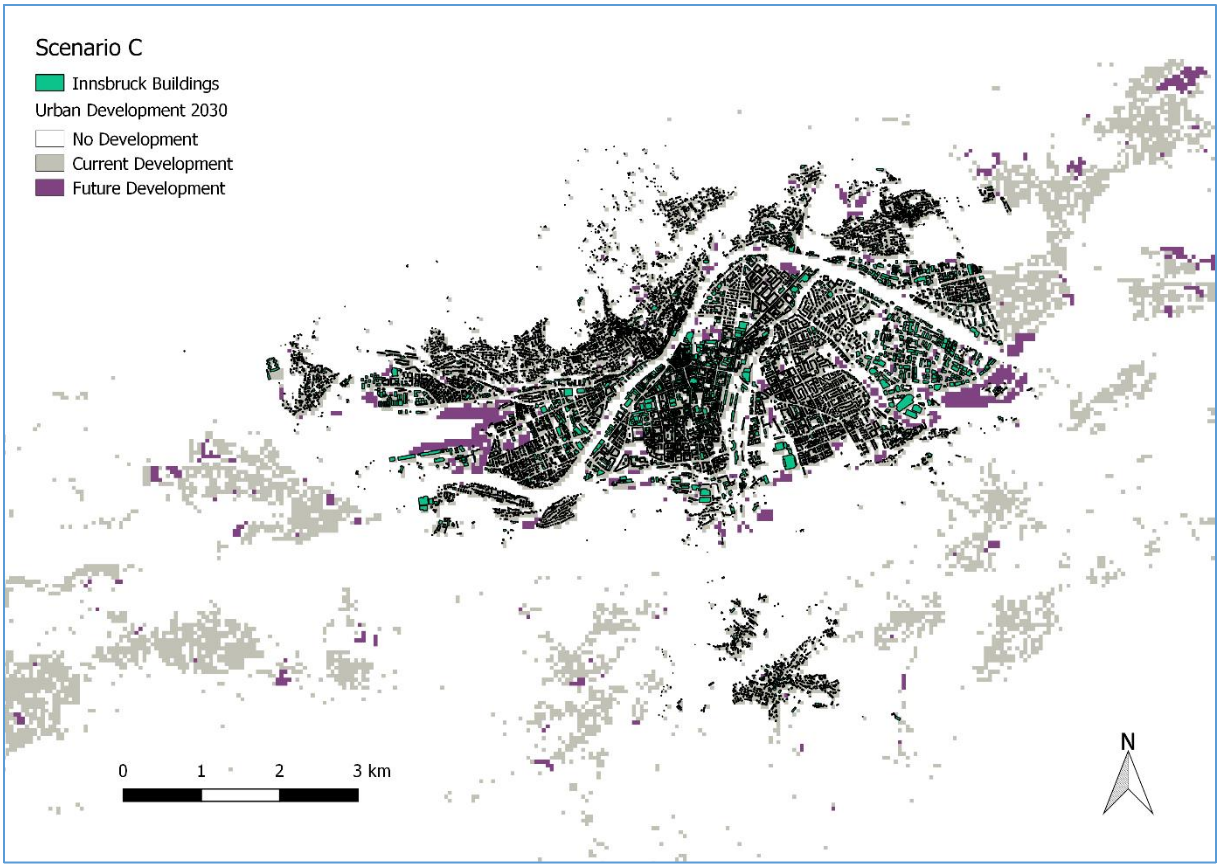

Figure 8.24: Scenario C cellular development.

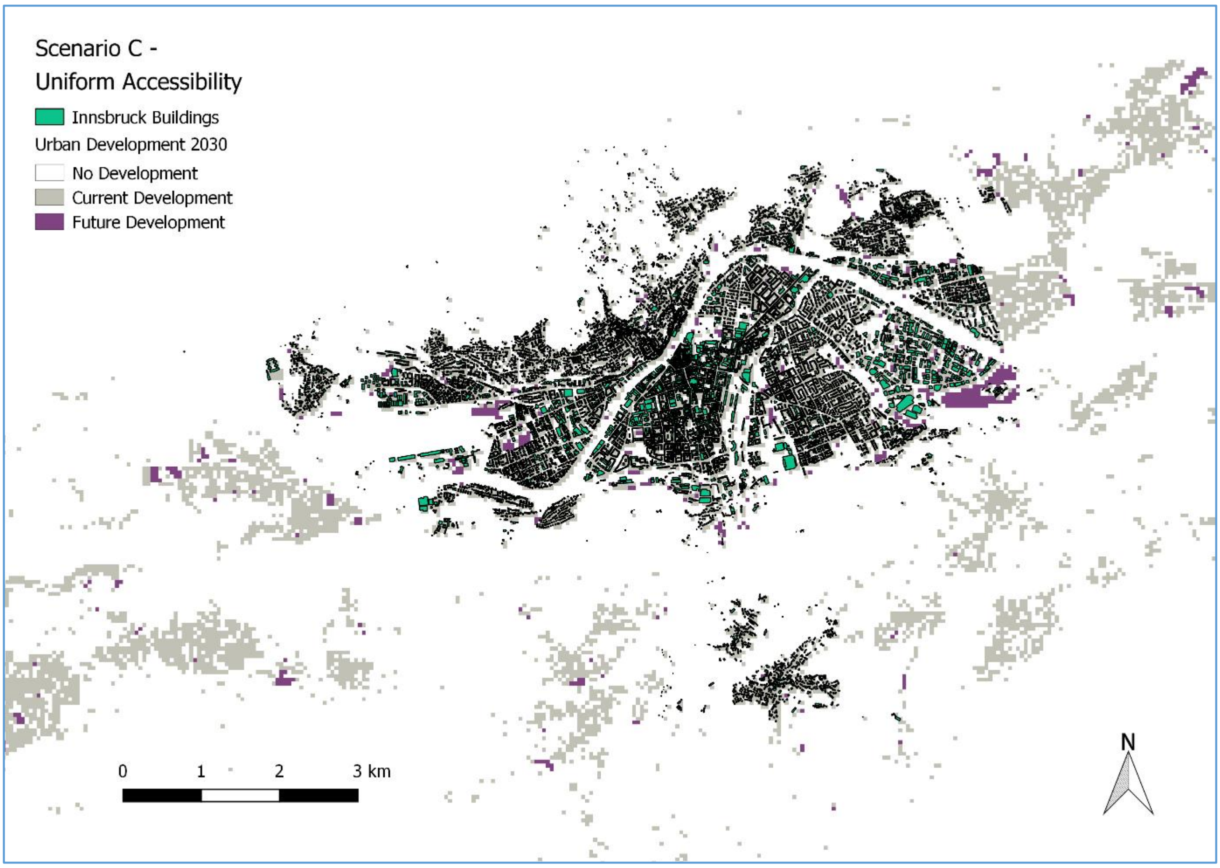

Figure 8.25: Scenario C uniform accessibility cellular development.

8.8 Summary

This chapter has described the application of the UIMF to a case study focussed on the city of Innsbruck and its surrounding region; the Inn Valley. Spatial policies set out in the ÖROKO were used to form attractors and constraints to the SIM; this was parameterised using observed data and driven using simple linear projections of employment and population for three scenarios. A single input network was used for TAM parameterisation which was modified using network processing to build access edges. A transport cost matrix was produced using network distance as a simple measure of cost which was then converted to an accessibility matrix input to the zonal model.

SIM outputs for employment and population were presented for the year 2030 to match the linear projection scenarios A and C. A further zonal population output was presented to indicate the possible development resulting from uniform accessibility. In all cases, the outputs demonstrate the correct operation of the model in disaggregating future development in accordance with spatial attractors. The UDM was parameterised with raster data to drive and constrain intra-zonal cell development in a consistent manner with the inter-zonal spatial disaggregation of the SIM. UDM raster outputs were described for development area identity, suitability based on input rasters and obtained using multi-criteria evaluation, and development area average suitability. The final outputs for UDM were cell development rasters for the year 2030 which correspond to the SIM outputs presented earlier; the cellular development pattern demonstrates coherence across modelling scales.

This chapter has provided evidence that the UIMF can be transferred from one study region to another and demonstrates its utility when using a minimal set of inputs. The outputs from each modelling stage show the correct operation of the models themselves, the coherence between models and across modelling stages. In the context of flooding and urban drainage, it is evident that the cellular output of the UDM could be used to drive modifications to existing drainage networks, and that the increase in impervious area resulting from development could be used in models of flooding. The model execution time for each stage in this case study is shown in table 8.3.

Model	**Execution time (seconds)**
TAM: network processing	45
TAM: cost and pathfinding	3
TAM: accessibility	1
SIM	2
UDM	6

Table 8.3: Innsbruck Case Study Model Execution Times

The following chapter discusses the findings of this research, including an examination of framework utility in the generation of results for the case studies presented in this chapter and for the GLA region (chapter 7).

Chapter 9. Discussion

9.1 Introduction

This chapter examines the utility of the Urban Integrated Modelling Framework (UIMF) beginning in section 9.2 by comparing the case studies presented in the previous two chapters. Section 9.3 presents further exploration of the Greater London Authority (GLA) region for a range of future spatial planning scenarios. Section 9.4 examines the applied utility of the framework. Densification scenarios are explored in section 9.5 demonstrating the flexibility of model group iteration and drivers in the UIMF. Section 9.6 describes the estimation process of model uncertainty and outlines supporting framework properties. Section 9.7 presents the modelling of both employment and population in the Urban Development Model (UDM) whilst section 9.8 proposes an alternative model configuration to spatially model decline in the UDM. Section 9.9 discusses the relationship between vector and raster data in the UIMF whilst section 9.10 presents work which further addresses issues of spatial scale. Section 9.11 discusses issues of simplification in the models implemented within the UIMF whilst section 9.12 examines operational issues including model interfaces and execution times. Finally, section 9.13 critiques the UIMF against the requirements for modelling with stakeholders identified in chapter 3.

9.2 Case Study Comparison

The case studies presented in the previous two chapters demonstrate the transferability of the UIMF by its application to different study regions. Chapter 4 described model parameterisation for the GLA region, a major global city, whilst chapter 5 applied the UIMF to the city of Innsbruck and its surrounding region; the Inn Valley, in the western Austrian state of Tyrol. A fundamental difference between the case studies was the level of detail in model parameterisation and therefore data requirements; the GLA region was parameterised using detailed spatial planning policy, multiple transport networks and generalised cost parameterisation whereas the Innsbruck study region used a single spatial policy attractor, a single transport network and distance-based cost. This demonstrates that the UIMF can make use of detailed parameterisation data, where available, and can also be parameterised to quickly produce a reasonable output using a sparse set of data acting as the basis for further model exploration; this transferability and scalability of model complexity using the UIMF are key to its utility and relative ease of application.

In terms of the Transport Accessibility Model (TAM), the GLA region case study models private transport as well as three modes of public transport (bus, light rail and heavy rail) using generalised cost to calculate edge weights for input to Dijkstra's algorithm. Private transport congestion charging is implemented using an input polygon of the charge zone along with parameters to apply charges and discounts to selected journeys. The detail of this approach to modelling transport accessibility is contrasted in the Innsbruck case study where a single network is used and edge weights for Dijkstra's algorithm are calculated using distance; this greatly reduces data requirements for the TAM and quickly produces a reasonable output at the expense of accuracy in the representation of transport accessibility for the region. The UIMF's ability to pre-process and modify transport networks for different study regions is underpinned by using an input spatial reference parameter; the GLA study region uses British National Grid (27700) while the Innsbruck study region uses Austria GK West (31254).

The number of zones is a fundamental parameter which defines any applied UIMF study region; the division of the study region into zones must be done so in such a way that empirical data to ground the models in observed reality can be obtained for each zone. In both case studies, zones are defined by areas for which census data can be used which splits the GLA region into 633 zones and the Innsbruck study region into 140 zones respectively. Both the number of zones, and the order in which zones are handled when loading and transferring data is crucial to the consistent definition of the study region across model scales. Transport models produce zone to zone origin-destination matrices of cost and accessibility; the Spatial Interaction Model (SIM) uses the transport accessibility matrix to disaggregate and output zonal population; and the UDM is driven by zonal population and employs class composition to group raster cells into the study region's zones.

The SIM disaggregates employment and population over all zones, and outputs zonal population which is further disaggregated to a fine-scale raster of expected future land use in the UDM; this disaggregation process operates on exogenously generated projection values over a fixed number of timesteps. There is considerable variation in the method of projection and the timeframe employed in each case study; the GLA region is driven by projected multi-sectoral employment and uses inverse activity to generate future population values based upon observed employment and population for 11 timesteps between 2000 and 2100; the Innsbruck study region uses three scenarios of linear increases specified separately for employment and population for seven timesteps between 2000 and 2030. This demonstrates the ability to drive the UIMF SIM with a wide range of projected employment and population data from varied sources; for any given set of input projected values, the SIM converts these values to ratios relative to the base year, these ratios are then applied to observed values of employment and population which are disaggregated across zones in the study region.

The process of zonal disaggregation is driven by a set of weighted attractors for both employment and population; this permits the use of a user-defined number of zonal attractors which can be enabled or disabled using the associated weights. Detailed spatial planning strategy as set out in the London Plan was used in the GLA region case study, defining both the set of spatial constraints which determine land availability for development, and the set of spatial attractors describing targeted development areas. The Innsbruck case study employed a much simpler approach where a single spatial constraint and a single spatial attractor were built from provided raster data to represent Austrian spatial development planning for the study region.

The UDM outputs a raster of expected land use which permits the fine-scale assessment of sub-zonal development and its interaction with spatially detailed models of climate hazard impacts and adaptation. Both case studies make use of input raster data to represent sub-zonal attractors and constraints which are consistent with the inputs to the zonal model. The comparison of case studies shows that this fine-scale detail can be tuned as required to provide the desired cellular representation of the study region; the rasterised GLA region is described by 100 metre cells arranged in 585 columns and 450 rows, while the rasterised Innsbruck study region is described by 50 metre cells arranged in 2380 columns and 1300 rows. Cells are grouped into development areas of a specified minimum size; the GLA case study sets this parameter to four to stipulate that only cells which can be grouped into contiguous blocks of four hectares with at least three other cells are considered for development. The Innsbruck case study sets this parameter to one allowing all cells to be considered which results in relatively small areas of new development; altering this parameter to represent different development policy would result in different fine-scale patterns of land use for assessment in terms of urban drainage and flooding.

9.3 GLA Scenarios

The applied utility of the UIMF extends far beyond the simple case studies discussed in the previous section which focussed on model parameterisation and transferability. To provide a more comprehensive demonstration of the application of the UIMF, a suite of previously generated scenarios combining spatial policy drivers with transport infrastructure investment is used to drive the UDM developed in chapter 3. The output values of expected population and density are used along with the raster of future development to provide detailed information which can be used in the comparative assessment of alternative policies in the context of climate change adaptation.

The intended application of the UIMF is to provide decision support, this includes the assessment of future climate impact risk when considering alternative policies for urban development. In this case, the risk of flooding will be considered which can be attributed to development taking place in flood-prone areas, extreme rainfall events and rising sea level. The tidal floodplain of the GLA region is potentially at risk from future flooding of the River Thames; this is due to projected increases in the mean sea level and associated increases in the frequency of surge tides.

Four land use scenarios are used representing extreme policy drivers, these are simulated using UIMF models to produce spatial outputs of land use, population and density for the year 2100 which permits the comparative assessment of flood risk across all scenarios. The scenarios were constructed relative to the development attractors and constraints of the zonal model along with accessibility attributed to current transport infrastructure, and to future scenarios of low and high infrastructure investment. These four narrative scenarios are as follows:

- **Baseline**
    - Development attributed to future increases in employment and population continues to follow current trends. Policies set out in the London Plan are extrapolated into the future along with the current level of transport infrastructure investment.
- **Eastern**
    - Development is concentrated along the eastern axis of the city towards the Thames estuary. This involves the redevelopment of brownfield areas and the assignation of significant employment and transport infrastructure investment to the east of London.
- **Central**
    - Development is focussed on central London where observed employment levels are highest to reduce greenhouse gas (GHG) emissions from transport.
- **Suburban**
    - Development is targeted outside of central London towards metropolitan centres and suburbs, supported by significant investment in transport infrastructure for satellite towns.

In each of these scenarios, existing constraints such as the greenbelt were adhered to, and the attractors identified in the London Plan were retained as principal drivers of urban change. Zonal population projections in the year 2100 for each of these scenarios are shown in figures 9.1 - 9.4.

Policy and population projections over this timescale have inherent uncertainties which result in a lack of stakeholder confidence in model outputs; however, rather than predicting development, the outputs are aimed at parameterising the extremities of policy projection using contrasting 'what-if' scenarios which can then be used to inform and explain the exploration of the scenario space i.e. by comparing the policy drivers and outputs for the central and suburban scenarios. The differing future patterns of zonal population resulting from a range of land use and transport planning scenarios provide a consistent basis upon which comparative analysis and implications can be drawn, therefore informing the decision-making process. In many cases, this zonally aggregated data is sufficient to inform debate; where this is not the case and more spatial detail is required, the outputs from the SIM can be used to drive the UDM to develop a raster output at the specified cell size.

Each of the four zonal population scenarios can be used to generate corresponding rasters of land use development using the UDM; these scenarios can be expanded upon by varying the raster inputs and parameters representing spatial planning policy. UDM land use rasters based on the zonal population projections for the four narrative scenarios are shown in figures 9.5 – 9.8; these outputs are at a resolution of 100m and use spatial policy drivers consistent with the corresponding SIM inputs. The increased spatial detail of the raster output of expected urban development allows fine-scale assessment of the possible implications of development policy such as vulnerability to flooding which may lead to the development of adaptation options.

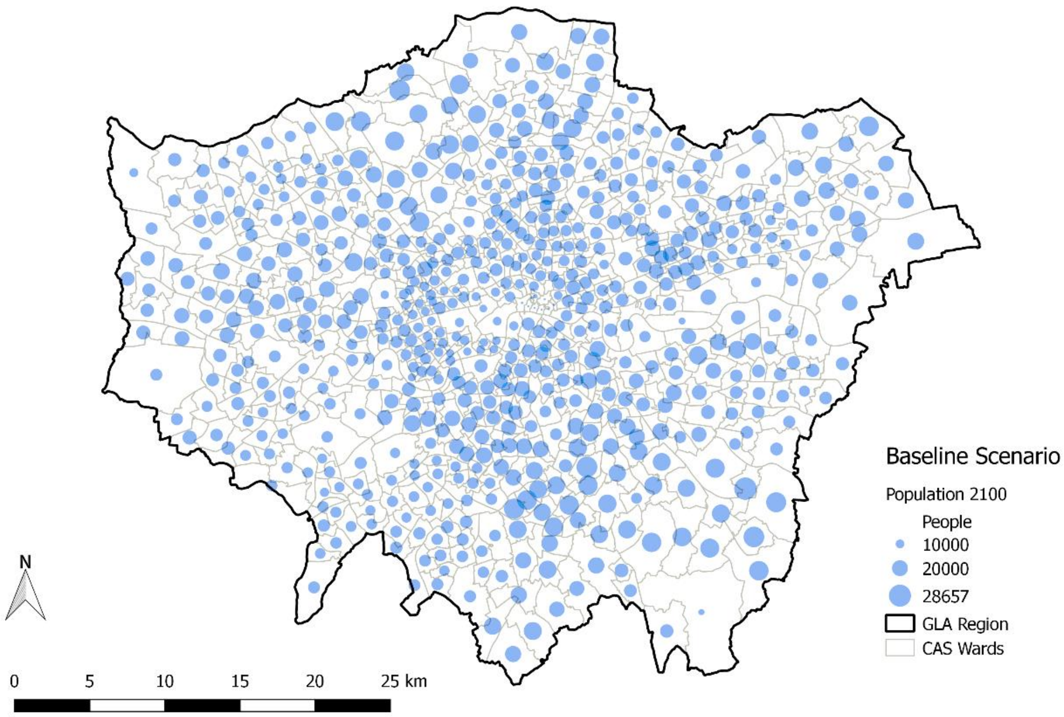

Figure 9.1: 2100 GLA population for baseline scenario output from zonal model.

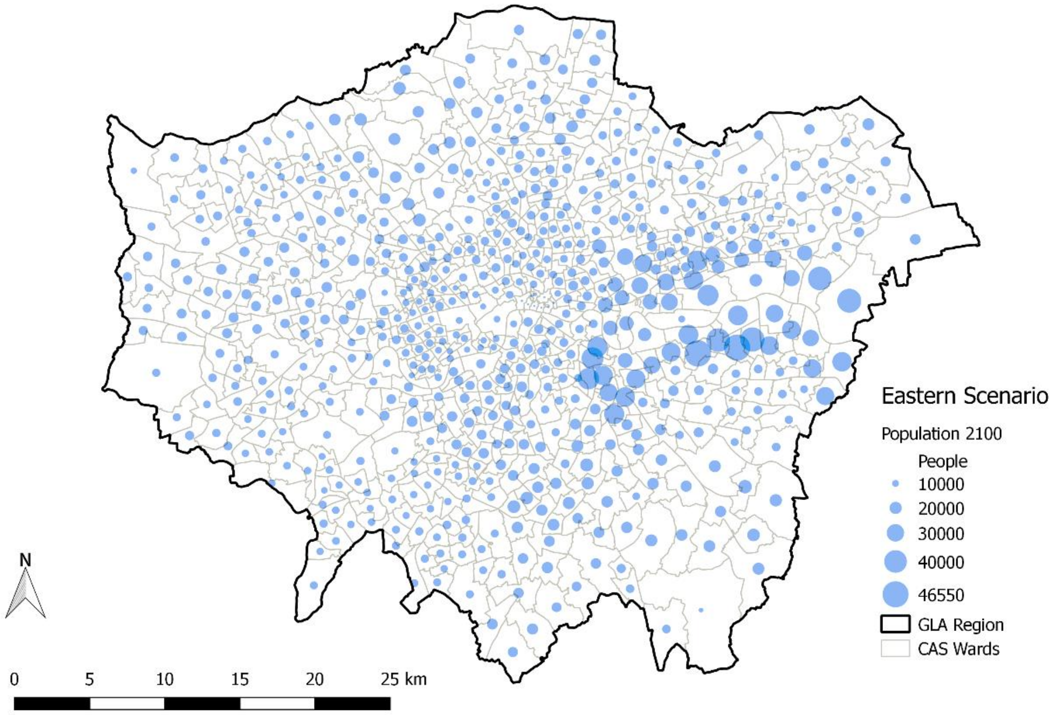

Figure 9.2: 2100 GLA population for eastern scenario output from zonal model.

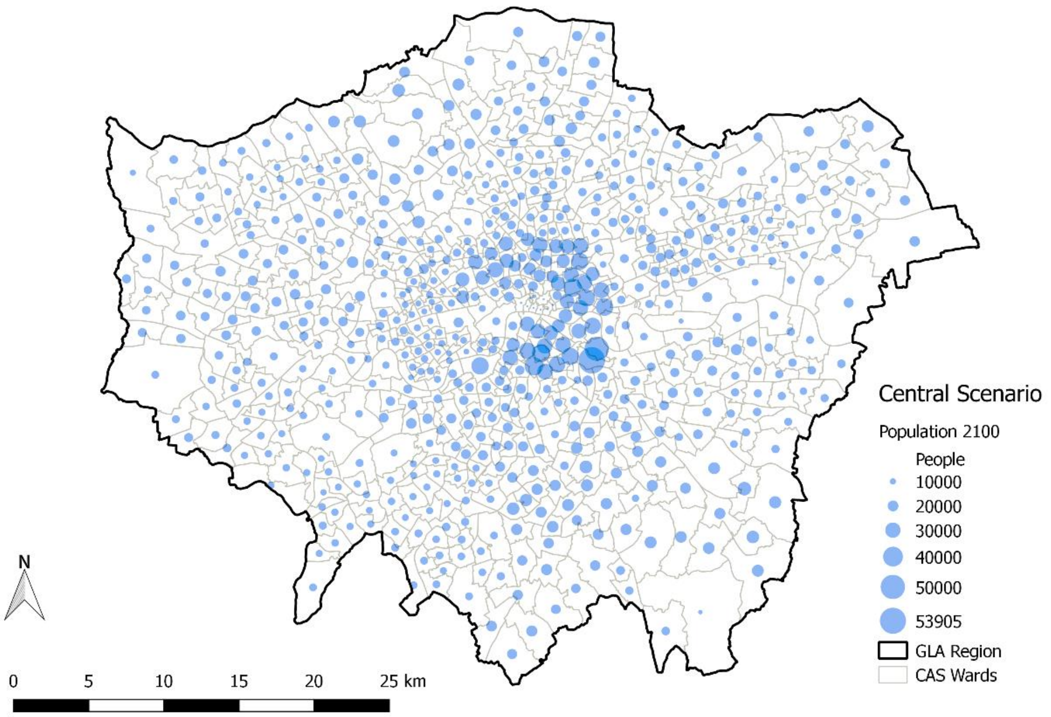

Figure 9.3: 2100 GLA population for central scenario output from zonal model.

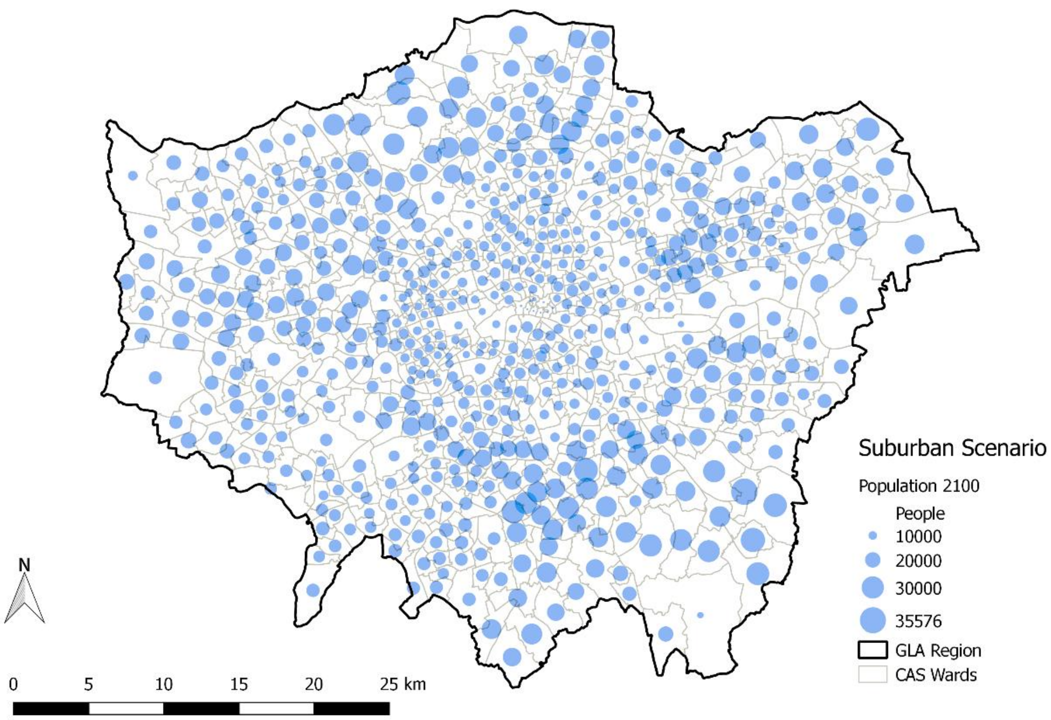

Figure 9.4: 2100 GLA population for suburban scenario output from zonal model.

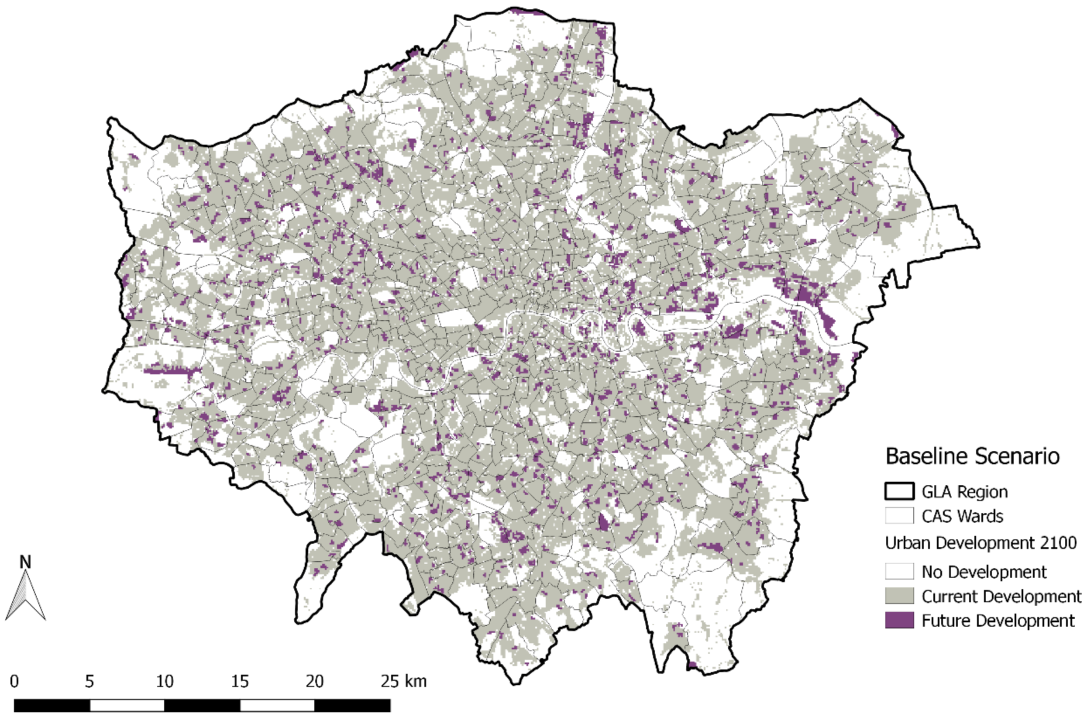

Figure 9.5: 2100 GLA land use development for baseline scenario output from UDM.

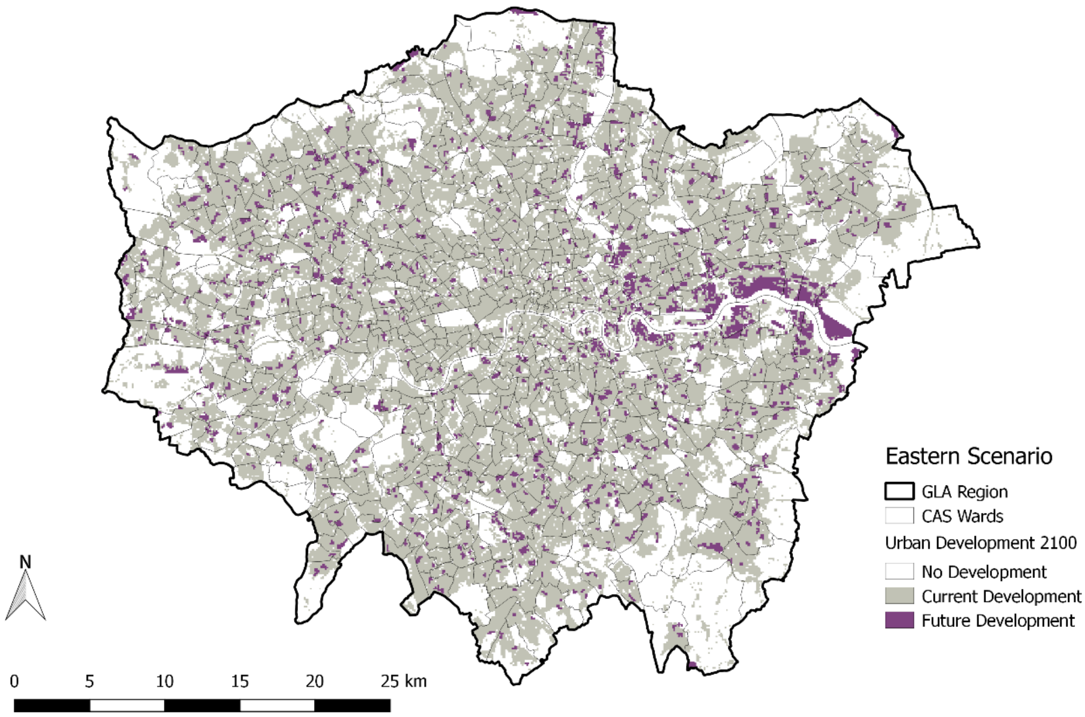

Figure 9.6: 2100 GLA land use development for eastern scenario output from UDM.

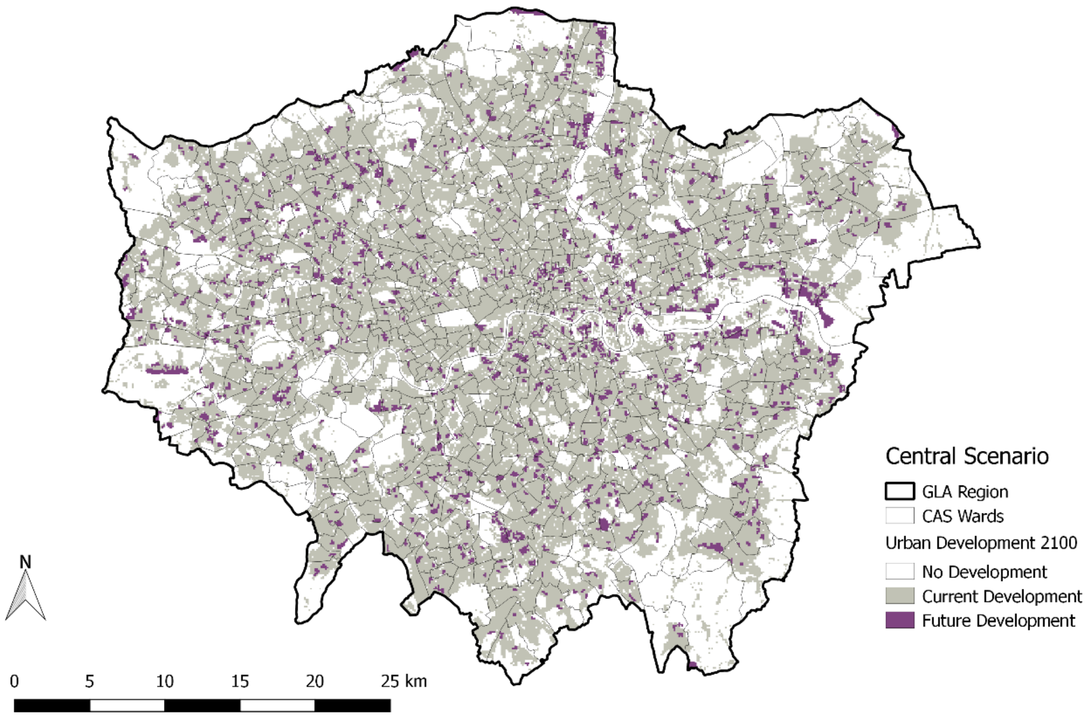

Figure 9.7: 2100 GLA land use development for central scenario output from UDM.

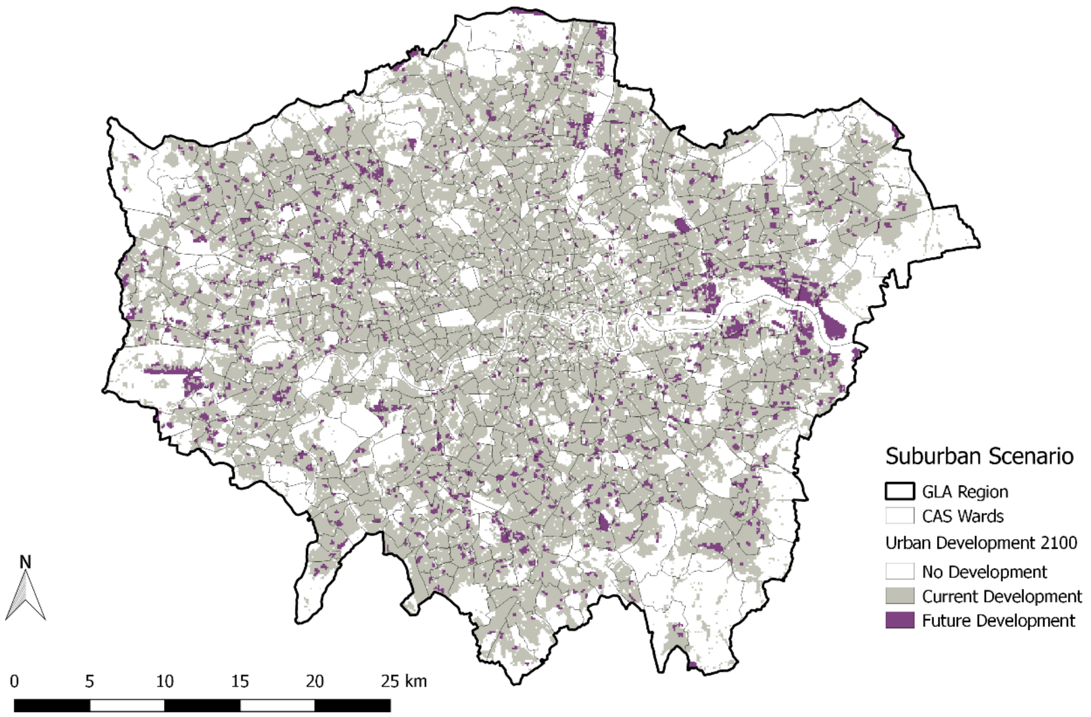

Figure 9.8: 2100 GLA land use development for suburban scenario output from UDM.

The UDM can also be used to calculate the density requirements of new development providing further information for impact assessment. The density of development in the UDM is made consistent with the SIM by using observed densities in both models; future development density is assumed to be the same as that observed unless policy drivers and projected population require densification to accommodate the increase. The required density adjustments reported by the UDM are included in the overflow data described in chapter 3. To make use of this densification reporting, the constraints used in the SIM and UDM are deliberately made inconsistent, removing constraints from the zonal model so that more land is available; the UDM then applies the complete set of constraints and calculates the density required to accommodate the projected zonal population increase according to spatial policy drivers. The required spatial densification to accommodate increased demand for population growth corresponding to each of the four scenarios is shown in figures 9.9 – 9.12. Table 9.1 summarises the implications of each of the four scenarios in terms of increased population density and new development in the flood plain.

Scenario narrative	Development in flood plain	Population densification (mean)
Baseline	+1697 hectares	114.50 people/ha
Eastern	+2160 hectares	106.30 people/ha
Centralised	+1552 hectares	123.10 people/ha
Suburban	+1503 hectares	115.30 people/ha

Table 9.1: GLA scenario summary.

Attractors and constraints in the UDM can be used to test a suite of policy options in response to future climate threats. Taking the eastern population scenario produced by the combination of spatial policy drivers as an example, the following three contrasting planning options are considered:

- Eastern planning scenario A: Remove the greenbelt constraint allowing more space for development away from the river as shown in figure 9.13.
- Eastern planning scenario B: Retain the greenbelt constraint and add a further constraint to prevent development in the floodplain as shown in figure 9.14.
- Eastern planning scenario C: Remove the greenbelt constraint but add the floodplain constraint as shown in figure 9.15.

Table 9.2 summarises the implications of each of the three scenarios in terms of increased population density and new development in the flood plain relative to the eastern scenario presented earlier. The UDM land use raster outputs along with maps of required densification for each eastern policy scenario are shown in figures 9.16 – 9.21.

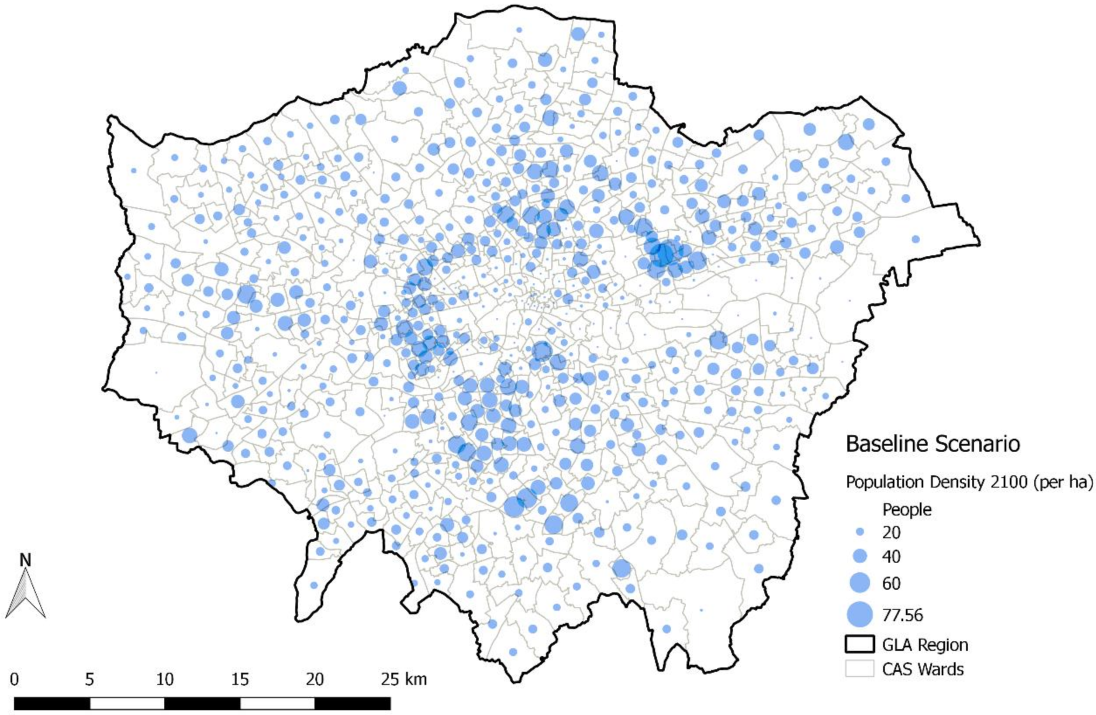

Figure 9.9: 2100 GLA population densification for baseline scenario output from UDM overflow data and aggregated to zones.

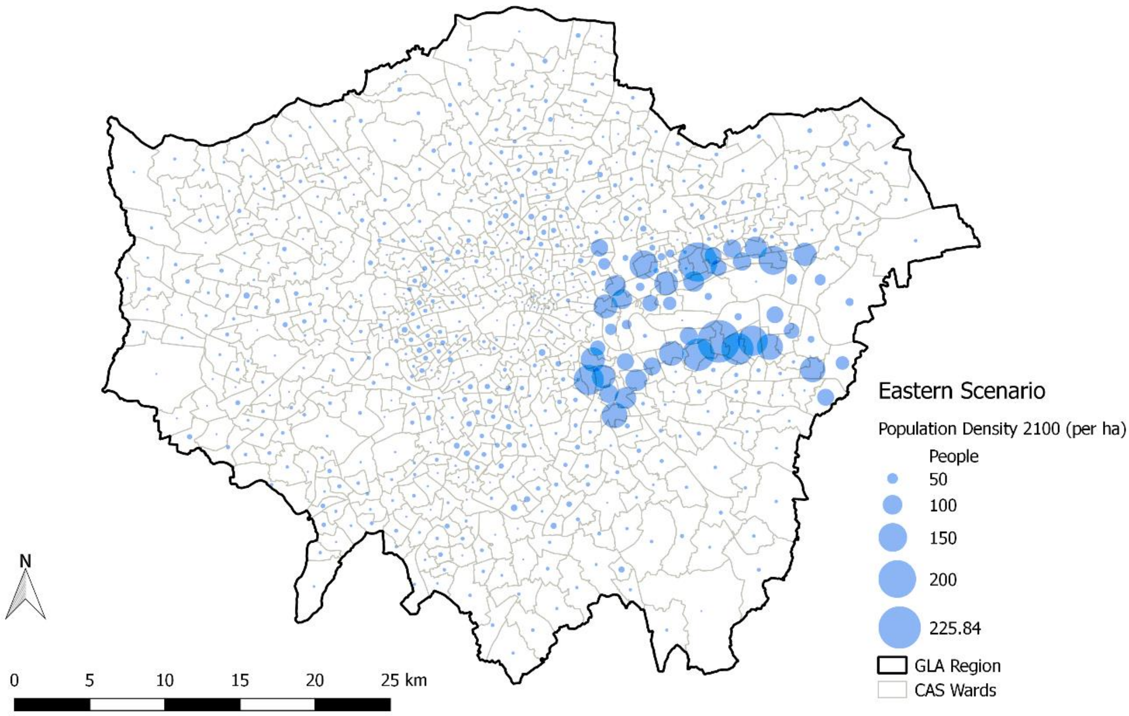

Figure 9.10: 2100 GLA population densification for eastern scenario output from UDM overflow data and aggregated to zones.

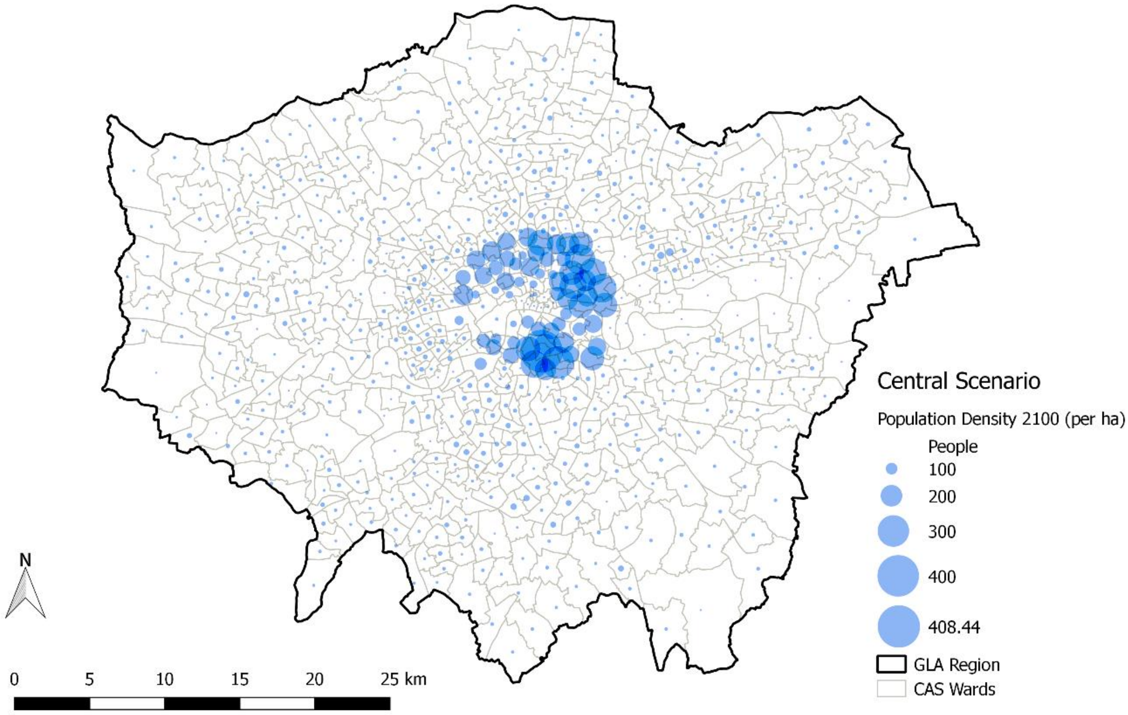

Figure 9.11: 2100 GLA population densification for central scenario output from UDM overflow data and aggregated to zones.

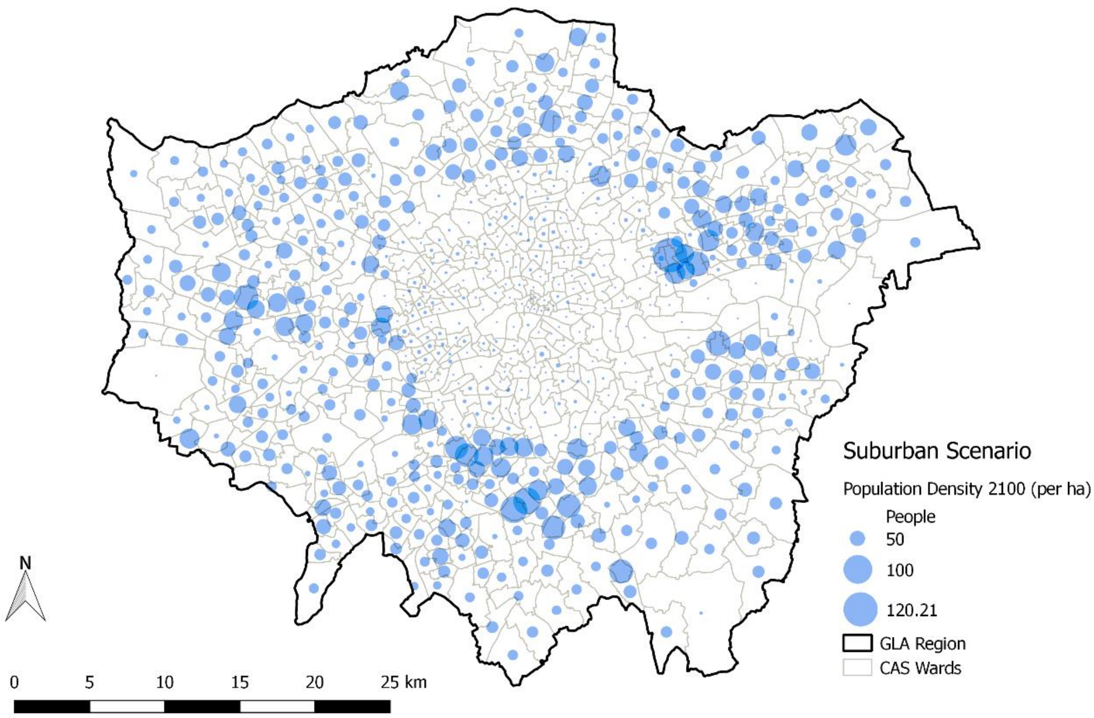

Figure 9.12: 2100 GLA population densification for suburban scenario output from UDM overflow data and aggregated to zones.

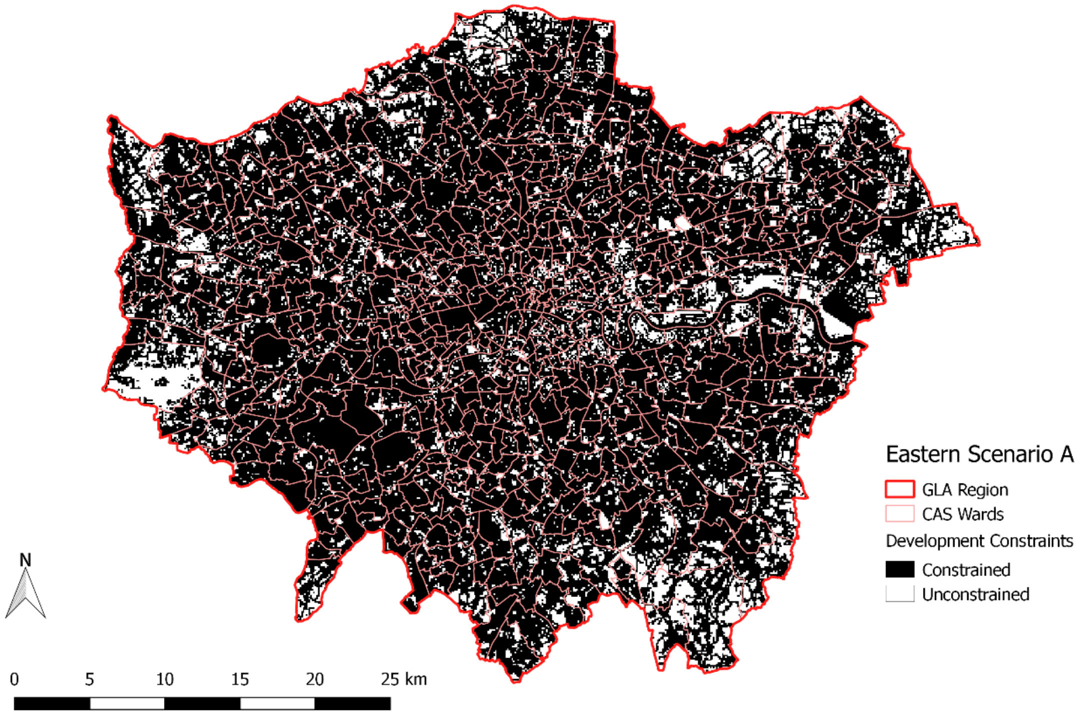

Figure 9.13: Constraint map for eastern scenario A input to UDM.

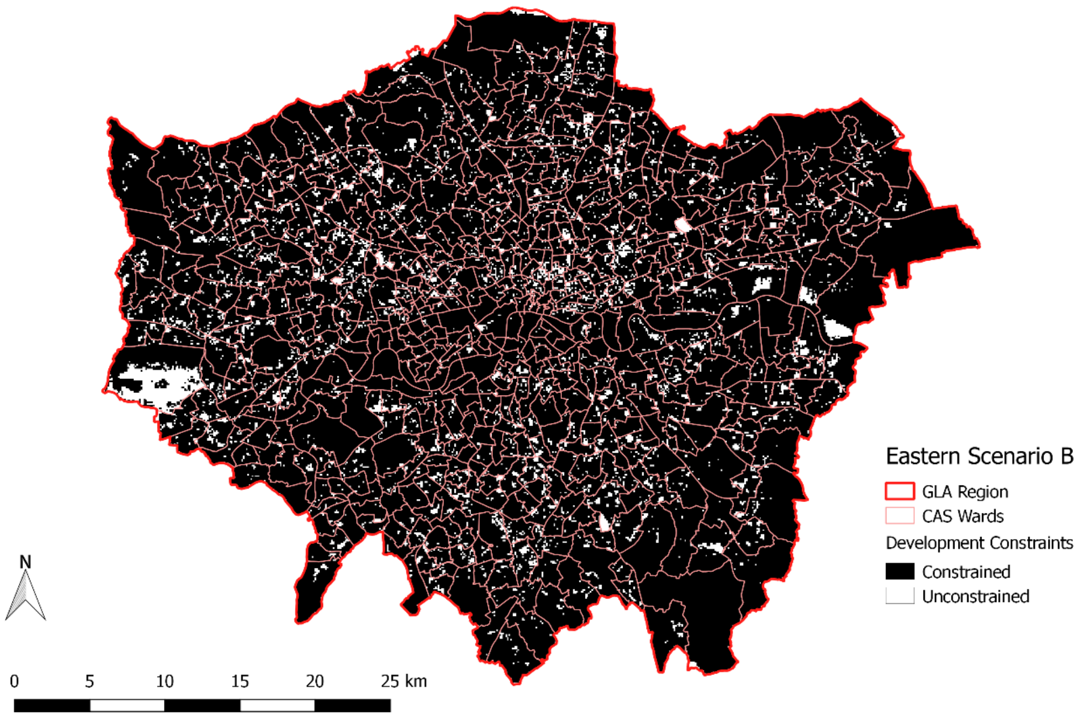

Figure 9.14: Constraint map for eastern scenario B input to UDM.

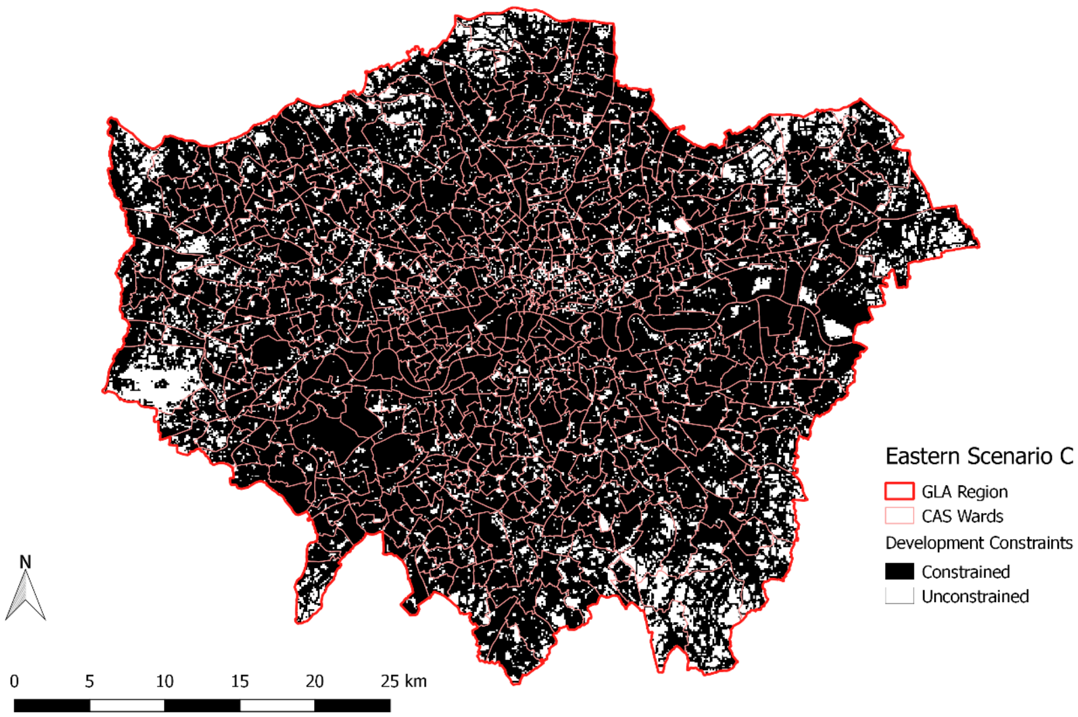

Figure 9.15: Constraint map for eastern scenario C input to UDM.

Scenario narrative	Development in flood plain	Development in greenbelt	Population densification (mean)
Eastern	+2160 hectares	0 hectares	106.30 people/ha
No greenbelt constraint, No floodplain constraint	+2219 hectares	+3646 hectares	104.29 people/ha
Greenbelt constraint, floodplain constraint	0 hectares	0 hectares	109.60 people/ha
No greenbelt constraint, floodplain constraint	0 hectares	+ 3783 hectares	107.62 people/ha

Table 9.2: Eastern population scenario policy options.

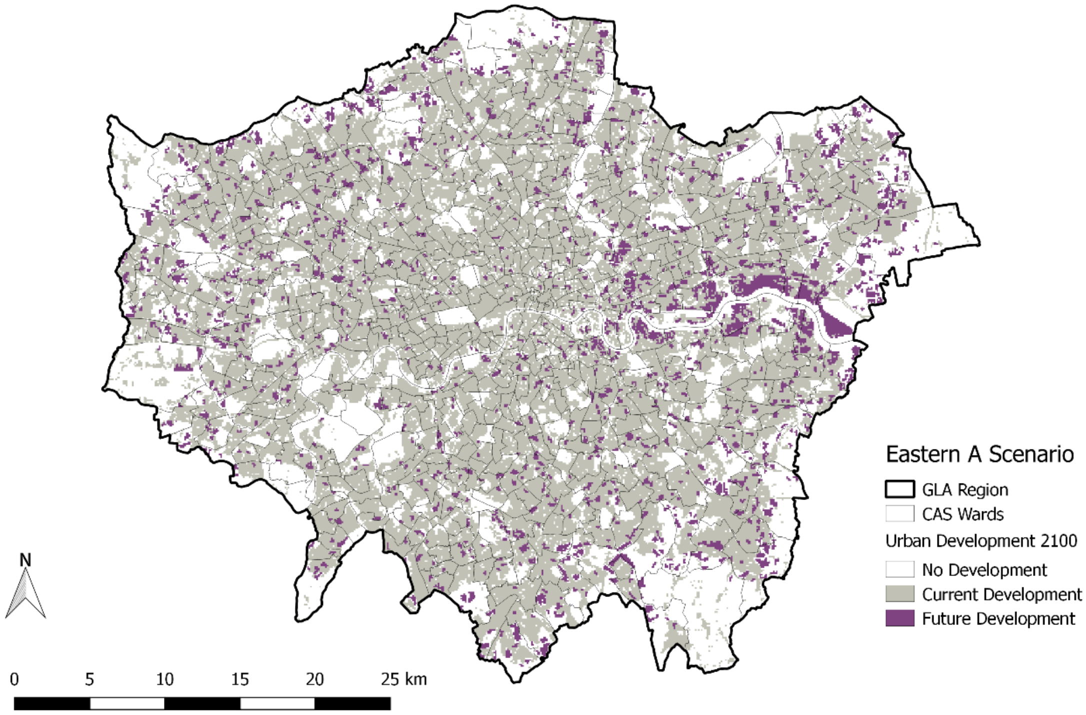

Figure 9.16: 2100 GLA land use development for eastern policy scenario A output from UDM.

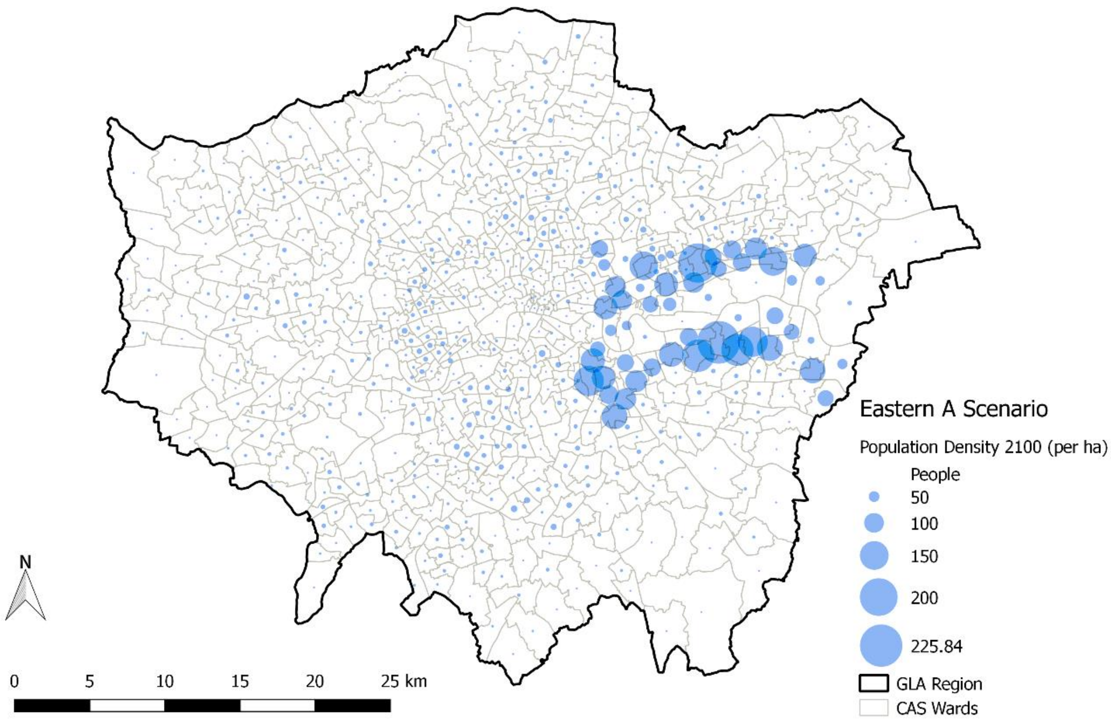

Figure 9.17: 2100 GLA population densification for eastern policy scenario A output from UDM overflow data and aggregated to zones.

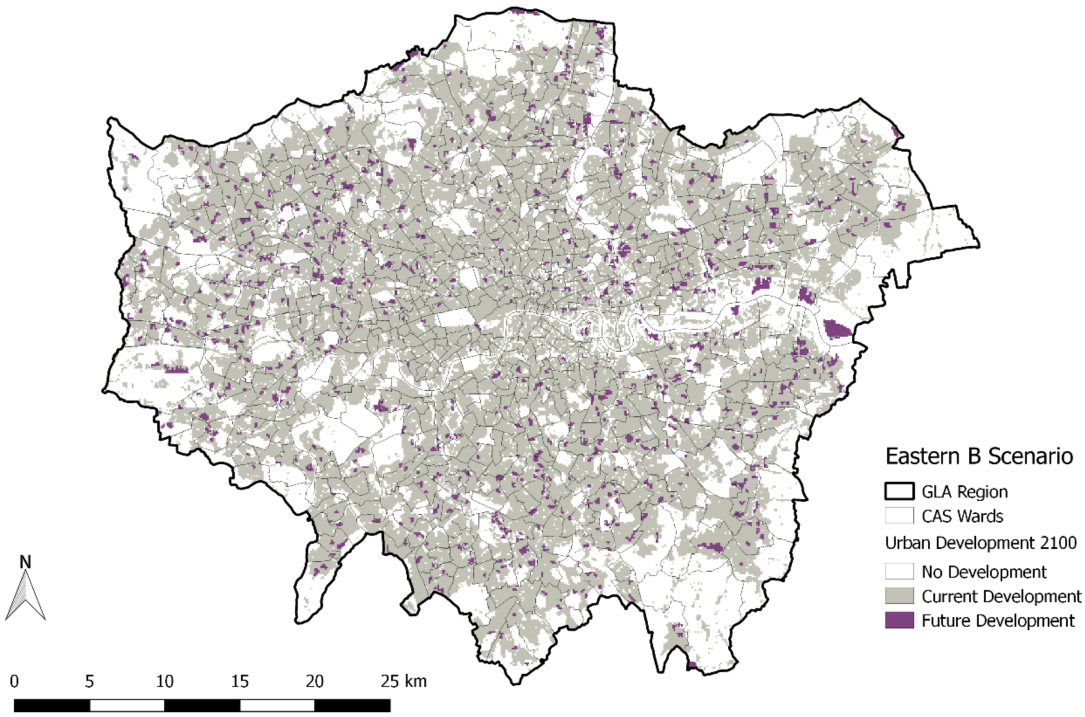

Figure 9.18: 2100 GLA land use development for eastern policy scenario B output from UDM.

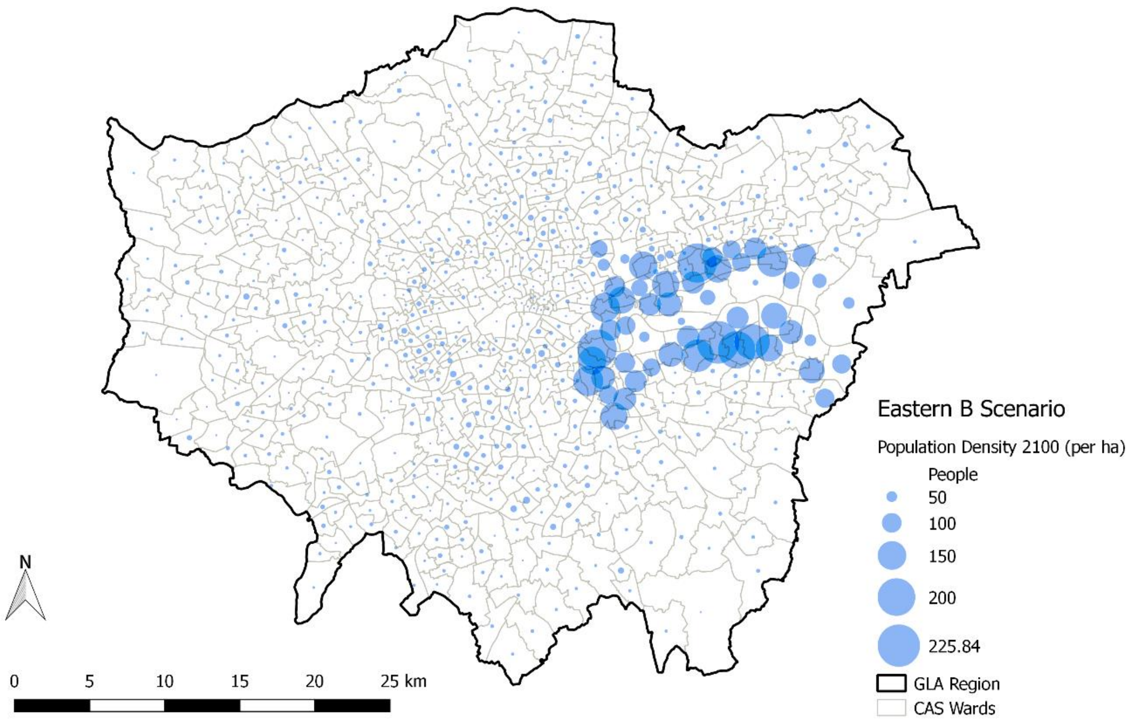

Figure 9.19: 2100 GLA population densification for eastern policy scenario B output from UDM overflow data and aggregated to zones.

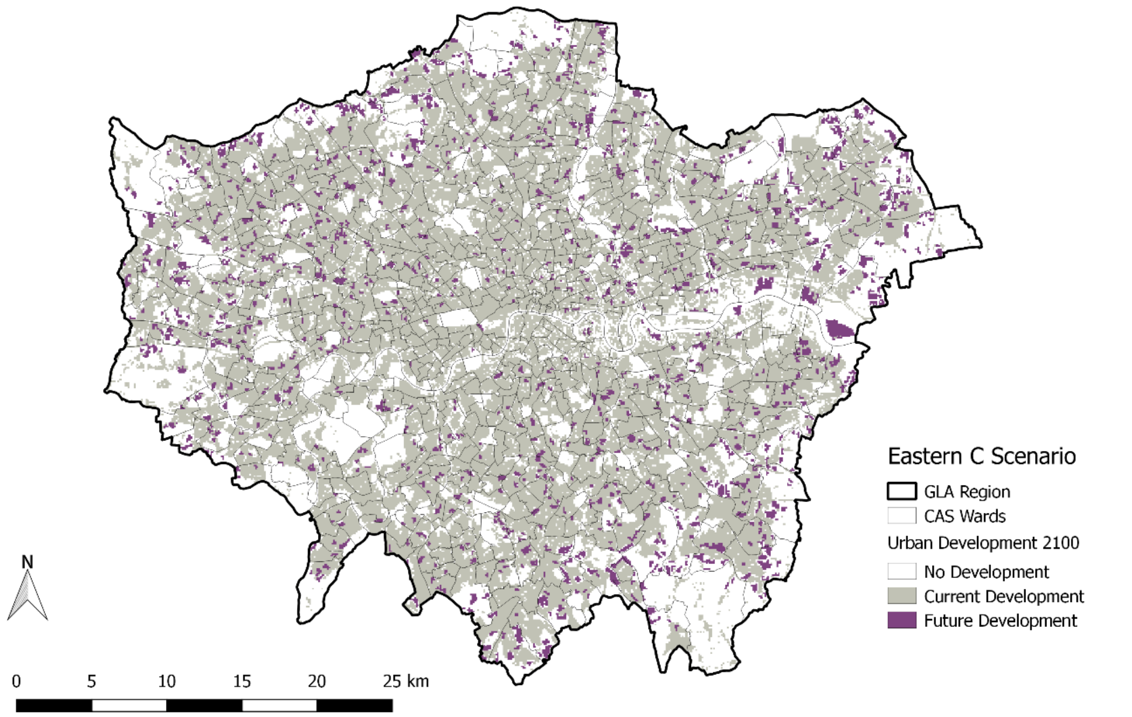

Figure 9.20: 2100 GLA land use development for eastern policy scenario C output from UDM.

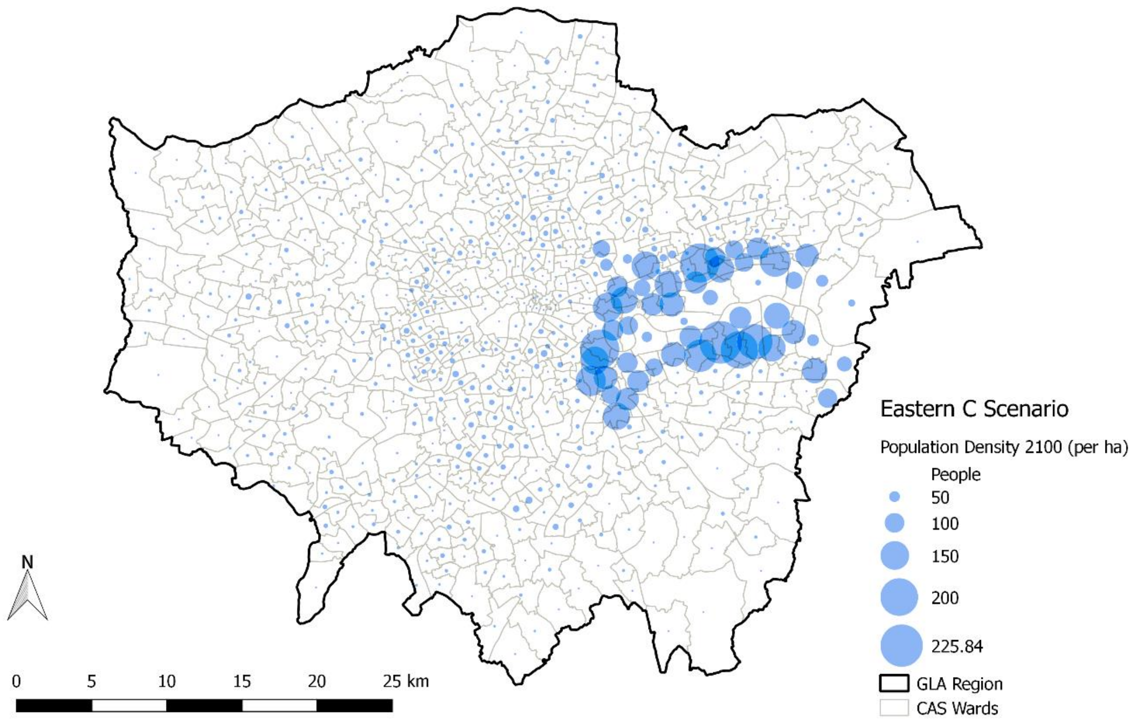

Figure 9.21: 2100 GLA population densification for eastern policy scenario B output from UDM overflow data and aggregated to zones.

This section has shown an approach to the integrated assessment of climate change impacts in urban areas using the UIMF. It has shown that by downscaling climate change impacts and socio-economic changes to a fine scale, understanding can be gained about the resulting patterns of vulnerability. As growing urban populations place pressure on cities to develop further, land in the floodplain or in the previously protected greenbelt may need to be developed - tensions between issues such as population density and flood risk are likely to increase. The linking of simulation modules which use spatial interaction modelling and cellular automata development projection with impact assessment is a powerful framework for exploring the implications of different planning policies and understanding their relative strengths and weaknesses.

9.4 Framework Utility

As presented in Chapters 7 and 8, the framework has been applied to two study regions whose spatial geography differs significantly; the parameters which define these regions, their spatial extent and the disaggregation of multiple factors including sectors of employment and development raster cell size are contrasted in Section 9.2. A variety of planning contexts are examined for the GLA region in Section 9.3, which distribute projected population and land use at two spatial scales for a range of spatial policy inputs. The results for the case studies and scenarios were generated by UIMF models using the same format for datasets across model applications; this demonstrates that the abstraction of common model properties is sufficient to transfer the framework models between study regions and planning contexts.

The cross-scale parameterisation of the SIM and UDM permits the fine-scale assessment of land use development which is consistent with spatial planning policy and transport planning. This fine-scale pattern of developed land is key to the interaction with climate models allowing the potential consequences of policy and planning to be analysed in terms of climate impacts which can then be used to inform a process of stakeholder dialogue and policy refinement.

In terms of the utility of the UIMF as a decision-support tool, results for the GLA scenarios described in Section 9.3 demonstrate the potential impact of spatial policy in terms of the area and population density of new development in the floodplain which increases risk. This climate threat was then factored into a range of alternative policies including the relaxation of the greenbelt constraint and the inclusion of a flood constraint which resulted in a further set of model outputs representing the competing pressures faced by urban planners; either develop in the floodplain, develop in the greenbelt or increase population density.

The Innsbruck case study presented in Chapter 5 demonstrates that the framework can operate using minimum data for simple study region parameterisation. The detailed land use patterns generated by UDM were used to assess potential impacts on urban drainage and flooding in wider research comparing simple approaches to land use modelling. The main objectives for the case study were achieved by demonstrating the transferability and usability of the framework while the results generated by simple model parameterisation form the basis for further work exploring policy scenarios and potential impacts for the Innsbruck region.

In summary, the generic transferability of the framework has been demonstrated which enables its application to a range of problems without further modification of the framework models themselves. The coherency provided by the framework in terms of model parameterisation and execution is essential to its utility and flexibility as a decision-support tool.

9.5 Densification

As described in section 7.3, projected employment and population for the GLA region could not be accommodated for all timesteps using observed density values. Rather than relaxing constraints and thus making more area available for development, a global density adjustment was made in which the density of new development in each zone was twice the respective observed value. This simplification did not account for attractor areas in the London Plan which are specifically targeted for densification; opportunity areas and areas of intensification (chapter 7.2).

Two alternative scenarios of exploring densification are considered, both of which require an input table of Boolean values identifying zones targeted for densification and involve the adjustment of density values so that both employment and population can be accommodated for all timesteps. In the first scenario (A), the densities of targeted zones only are increased leaving non-targeted zones at observed values; since this could lead to implausibly high densities in targeted zones, a second scenario (B) is considered in which targeted zones use provided density values while the densities of non-targeted zones only are increased.

In each scenario, design features of the UIMF described in chapter 3 are used to automate and control the densification scenarios. As shown in figure 9.22, the zonal model and two driver models are placed into a model group which is iterated over using the forced-conditional option by setting $n\_iterations$ to -1 i.e. the group is run until the stop condition is reached. The first driver model controls the density adjustment process providing modified values which are input to the SIM, while the second driver model stops group iteration when employment and population can be accommodated for all timesteps.

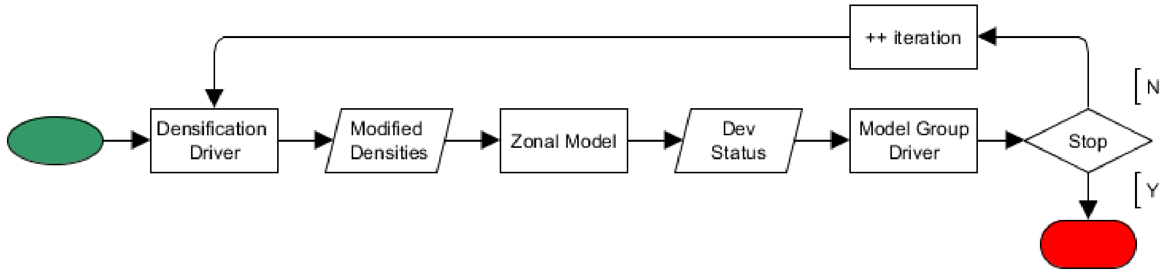

Figure 9.22: Densification scenario model group.

As shown in figure 9.23, the densification driver model takes the tables of observed density values, $\boldsymbol{D}_{(\boldsymbol{z})}^{\boldsymbol{obsrv}}$, for employment and population as inputs. The corresponding table names specified as inputs to the SIM are changed to point to temporary tables, $\boldsymbol{D}_{(\boldsymbol{z})}^{\boldsymbol{temp}}$, created and modified by the densification driver model in each iteration.

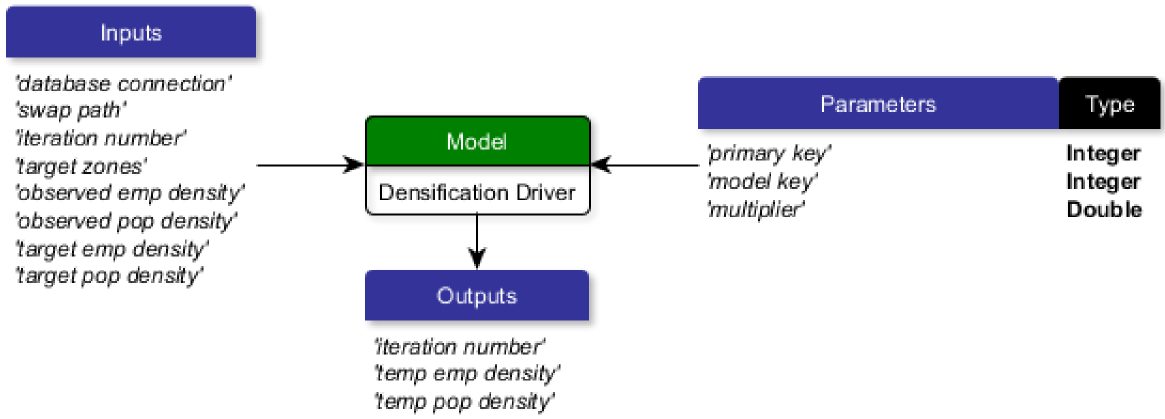

Figure 9.23: Densification driver model inputs, outputs and parameters.

The table of Boolean values describing target zones for densification, $\boldsymbol{T}_{(\boldsymbol{z})}$, is supplied along with target densities, $\boldsymbol{D}_{(\boldsymbol{z})}^{\boldsymbol{target}}$, for use in scenario B. To drive the adjustment of density values, the densification driver must keep track of the number of model group iterations. This integer is initially set to zero and is stored in a table named as both an input and an output to the densification driver model; the value is read from the table, incremented at the end of the model script and written back to the table. The granularity of density adjustment is controlled by the multiplier parameter, $\boldsymbol{mult}$, which is used along with the iteration number, $\boldsymbol{iter}$, to generate temporary values based on observed density inputs.

In scenario A, the densities of target zones, where $T_{(z)} == 1$, are incrementally increased from observed values according to equation 9.1 whilst non-target zones, where $T_{(z)} == 0$, are held at observed values as shown in equation 9.2. In scenario B, the densities of target zones, where $T_{(z)} == 1$, are set to the provided target values according to equation 9.3 whilst non-target zones, where $T_{(z)} == 0$, are incrementally increased from observed values as shown in equation 9.1.

$$D_{(z)}^{temp} = D_{(z)}^{obsrv} + \left(iter \times mult \times D_{(z)}^{obsrv}\right) \quad \text{(Eq. 9.1)}$$

$$D_{(z)}^{temp} = D_{(z)}^{obsrv} \quad \text{(Eq. 9.2)}$$

$$D_{(z)}^{temp} = D_{(z)}^{target} \quad \text{(Eq. 9.3)}$$

The densification driver model should write all inputs, outputs and parameters to a metadata table so that the SIM can access and stitch this information, before its own, into the chained model metadata; this process is automated by setting the SIM parameter is_driven to 1 and providing the name of the driver model.

The model group driver, as shown in figure 9.24, is responsible for ending group iteration when employment and population can be accommodated for all timesteps. The inputs are combined to form the fully qualified name of the development status .csv file which is output from the SIM to the designated folder used to transfer model data to and from database tables.

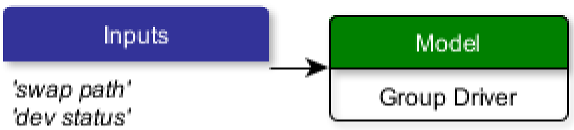

Figure 9.24: Densification scenario model group driver.

The development status .csv file output from the SIM C++ class documents whether employment and population were accommodated in each timestep. This is determined when handling area capacity corrections i.e. where there is excess employment or population in set Z1 zones which cannot be accommodated by spare capacity in set Z2 zones. The file consists of Boolean values for employment and population in each timestep where a value of 1 indicates that there was no overflow. The model group driver script simply searches the file for values of 0, if any are found then group iteration is continued by returning a $model.stop$ value of false; if none are found then group iteration is stopped by returning a $model.stop$ value of true.

The techniques presented in this section permit the detailed exploration of spatial densification scenarios where the available area for development is held constant and projected employment and population must be allocated land. Using the setup for scenario A, it can be established whether the densification of target zones only can accommodate future land use within plausible density bounds. Using the setup for scenario B, the impact in terms of required densification on non-target zones can be found when specific density values are provided for target zones. A range of different target zones and associated densities could be tested to generate a portfolio of options to inform densification policy.

9.6 Uncertainty

Long-term projections of land use are inherently uncertain even when using complex models whose aim is to predict with some degree of accuracy. The UIMF provides the basis for a simple, fast and auditable approach to understand the scale and implications of uncertainty which can help stakeholders make better informed choices.

Consider the case of vehicle operating costs used in the computation of generalised cost for private networks described in chapter 5; the coefficients used are based upon a profile of vehicles, their fuel efficiencies, and fuel prices. This mixture of vehicles is subject to change due to initiatives to increase fuel efficiency, transition to electric vehicles and reduce GHG emissions, whilst the price of fuel is highly volatile. It was demonstrated in chapter 8 that the zonal disaggregation of population is strongly influenced by accessibility to employment via the matrix of transport accessibility; uncertainties in fuel prices can cascade through UIMF models all the way from generalised cost to the fine-scale raster of land use development in the UDM.

Confidence in the output or response of any computational model is restricted by uncertainty in the model inputs such as errors or bias in data, and by uncertainty in the concepts of the model itself including the adopted simplifications from reality. To form scientifically robust arguments based upon such a model, it is necessary to assess the uncertainties linked to its response and determine the level of confidence in the model. In cases where the precision of a model is deemed to be unsatisfactory i.e. when Uncertainty Analysis (UA) returns an estimated model response distribution which fails to meet a given standard, it is desirable to be able to apportion responsibility to individual input factors to guide improvements in the modelling process. This quantification of individual input uncertainty contributions to the variation of overall model response is achieved via Sensitivity Analysis (SA).

UA provides an evaluation of model output uncertainty as a function of the uncertainties associated with its inputs. In UA the model in question is considered to have a single output variable, Y, and k input variables, $X = (X_1, X_2, \dots, X_k)$, which capture the total potential for uncertainty from both data-based and model-based factors in terms of stochastic variables with a predefined Probability Density Function (PDF). Equation 9.4 shows that the link between model response and its inputs is described by a mathematical function of the computational model, $f()$, mapping from the k-dimensional space of the inputs to the single output variable, Y. UA quantifies the effect of input uncertainties and facilitates the estimation of the output variable's PDF using an expected value (equation 9.5) and variance (equation 9.6); this provides an assessment of the precision of the model (Crosetto et al., 2000).

$$Y = f(X_1, X_2, \ldots, X_k) \quad \text{(Eq. 9.4)}$$

$$\hat{E}(Y) = \frac{1}{N}\sum_{i=1}^{N} Y_i \quad \text{(Eq. 9.5)}$$

$$\hat{V}(Y) = \frac{1}{N-1}\sum_{i=1}^{N}((Y_i - \hat{E}(Y))^2 \quad \text{(Eq. 9.6)}$$

The development of an uncertainty framework will be addressed in future research based upon a four-step Monte Carlo (MC) technique to estimate uncertainty as follows:

Step one: Allocate PDFs to model inputs, X_i; this process should be informed by the nature of each input in terms of its potential for uncertainty to generate an appropriate range;

Step two: Generate a sample set, $N(X^j, j = 1, \ldots, N)$, from the input factors' distributions using some form of random sampling technique;

Step three: Execute the model for each sample point, X^j, using the generated input values to produce an output value for each iteration;

Step four: Carry out analysis on the response variables,Y^j. Estimate the expected value (equation 9.2), variance (equation 9.3), and produce response variable histogram.

Assuming step one has been carried out to generate an input distribution for X_i, steps two and three involve executing the model with randomly sampled inputs, X_j, from that distribution to produce a set of outputs, Y_j for analysis. This process could be automated using model group iteration and a model driver to modify model inputs, X_j, execute the model $f()$, and store model outputs, Y_j, along with metadata for all required iterations. The time taken to compute the MC uncertainty estimation on any given system is almost entirely governed by the execution time of the model under test; the use of high-performance C++ models in the UIMF which are wrapped by SWIG and executed from Python scripts minimises this computational cost and permits the generation of output values from multiple model executions in a short timeframe.

9.7 Multiple UDMs

As described in chapter 8, the raster of land use change attributed to population development which is output from the UDM has utility when examining localised effects of surface sealing and drainage network modification. The only relevant output which reflects land use change for both employment and population is at the zonal level where the change in available area within each zone could be directly interpreted as a change in the impervious area for each zone; this output may not be sufficient when considering flooding and drainage at a highly localised level so a raster of land use change attributed to employment development could be generated to provide a more spatially detailed output.

A further rationale is that the way in which the UDM is used to produce snapshots of future population development relative to the defined base-year is not consistent with the allocation of land in the SIM; for each timestep in the SIM, land is allocated first to employment change, then to population change, where area availability is reduced proportionally for each zone by each allocation.

Consider an arrangement of multiple UDMs which map the change in both employment and population over all simulated timesteps to produce a more detailed account of land use change associated with urban growth. For the purposes of simplification, only positive change is considered; section 9.7 discusses the possible representation of negative change in the UDM. Features of the UIMF described in chapter 3 are used to control and coordinate the multiple UDMs used in this modelling arrangement which is shown in figure 9.25. This involves the update of raster data between UDM runs to reflect changes in development along with model feedback across iterations to synchronise UDM land use change for employment and population development with values generated by the SIM.

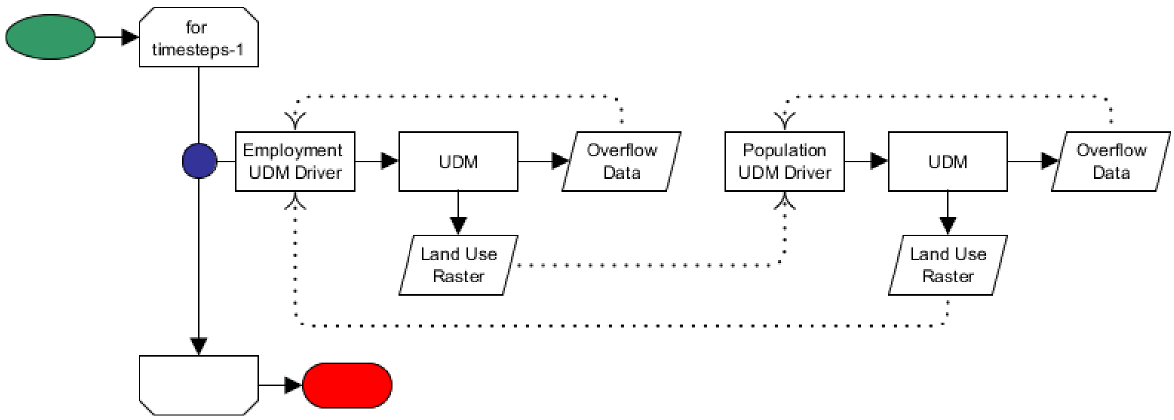

Figure 9.25: Multiple UDMs model group.

As shown in figure 9.26, UDM driver models take the previously generated development raster, udm_raster, as an input, this raster data must be processed by the respective UDM driver model to provide updated raster inputs referenced in udm_raster_tables to the next instance of UDM. chapter 3 describes the format of the cellular development raster output from the UDM as follows:

- -1: represents no data values.
- 0: represents no development i.e. constrained areas.
- 1: represents current development.
- 2: represents future development.

The UDM driver models are responsible for using this output raster data to generate updated rasters of current development, development proximity and constraint for use by the next instance of UDM. Converting future development to current development can be achieved using the framework raster function $Change\_Ref\_Val()$ to convert raster cell values of 2 to values of 1; this produces an updated current development raster from which an updated raster of development proximity can be generated. The modified current development raster can then be used to update the constraint mask by first using the framework raster function $Not\_Boolean()$ to convert values of 1 signifying current development to values of 0 signifying constraint, then by combining with the current constraint raster using the framework raster function $Combine\_Boolean()$. After processing this raster data, the UDM driver models must ensure that the correct rasters are present in the folder assigned to swap data, and that they are correctly named in the corresponding input raster tables.

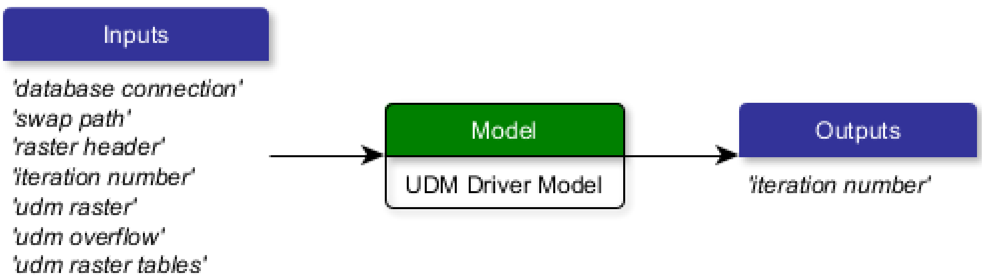

Figure 9.26: UDM driver model inputs, outputs and parameters.

Aside from managing the update of raster data between UDM instances, the UDM drivers must adjust values of employment and population change to ensure consistency with the SIM. An initial idea would be that since the available area in the SIM is calculated with reference to available cells in the initial constraint raster input to the UDM, that the available area should be updated after each UDM run; however, this would lead to an inconsistent linkage between modelling scales since the UDM must assign a cell for development when presented with positive change, even if the cell is only partially occupied. In the interest of forming a working solution, UDM overflow data is used in such a manner that partially occupied cells from the previous run are filled up before new cells are developed in the current run; this negates the inconsistency between modelling scales and permits this modelling arrangement to function in accordance with the values generated by the SIM.

The method presented in this section permits a more detailed exploration of the temporal and spatial distribution of land use change attributed to development for both employment and population. This results in a more detailed model of land use change which could, for example, be used to examine the localised effects of surface sealing and urban drainage network modification.

9.8 Decline

The application of most urban models is restricted to scenarios of urban expansion and the associated land use demands of future development. For the GLA region presented in chapter 7, the total employment decreases between 2080 and 2100; since, in this case, the population was linked to employment via inverse activity, the population also decreases over the same timeframe. For simplification, the decision of how to handle land use for declining employment and population change can be reduced to a binary choice between retaining this previously occupied land or releasing the land for future development; the UIMF SIM adopts the latter approach.

As described in chapter 4, the UIMF SIM can spatially disaggregate employment and population where the change in either value for the current timestep is positive or negative. In the case of employment, reciprocal attractor values derived from the input weighted attractors are used to proportionally remove employment from zones. For population, reciprocal values based on both input weighted attractors and accessibility to employment are used to disaggregate population decline across zones. In both cases, negative change assigned to a zone results in land being given back to the zone in accordance with the zone's specified development density.

By giving back available area to declining zones, the SIM assumes that this newly vacated land and the buildings upon it can either be repurposed or is demolished allowing new development to take place. Clearly, the repurposing of existing building stock is not always feasible and is dictated by the differing requirements of housing and across employment sectors. In many cases, demolition is likely which incurs extra development cost since, in effect, the land would be classified as brownfield; however, the use of brownfield sites for new development is often incentivised as part of sustainability policy as in the London Plan described in chapter 7.

The UDM, as described in chapter 6, considers positive population change only, and is generally used to provide a view of future land use development relative to population in the base year. In most cases this results in positive change but section 9.6 describes how land use attributed to change in both employment and population can be output for each timestep relative to the previous timestep to provide a more detailed picture; in this case the inability of the UDM to model decline and give back available area for development would be inconsistent with the function of the SIM.

The UDM could be used to output a fine-scale representation of land reuse, repurposing or demolition for zones assigned negative change by the SIM; this could be achieved by the following adjustments:

1. The UDM calculates the number of cells required for development in each zone based on the amount of positive change and the development density. This number is converted from double precision to an integer using a $ceiling()$ function to ensure that a sufficient number of cells are developed to accommodate the positive change; this generally results in some excess development in which the final cell is only partially occupied. Assuming negative change assigned to a zone is identified and used to set a Boolean variable, the negative change is first switched to a positive value using an $abs(change)$ function. The number of cells to free is calculated as before but using a $floor()$ function for the conversion from double precision to integer to ensure that the total area removed by all cells does not exceed the assigned zonal decline i.e. each cell to free is completely empty.
2. UDM development can only be accommodated by unconstrained cells specified by a value of 1 in the input constraint raster; further to this, unconstrained cells are grouped into development areas of a specified minimum size. To configure the UDM to remove rather than assign development, the constraint raster should be replaced by the raster of current development in which values of 1 indicate potential cells for removal.
3. The UDM uses multi-criteria evaluation to create a raster of development suitability based upon the input constraint mask and a set of rasterised weighted attractors which are combined to determine each cell’s suitability for development. These suitability values are averaged for groups of cells in development areas and dictate the order in which both groups of cells and individual cells are developed within each zone. The

Multi-criteria evaluation function can be used to create a raster of reciprocal suitability in a similar manner to the zonal model's reciprocal attractors so that the same set of rasterised weighted attractors which determine suitability could be used to determine unsuitability; however, it is likely that a different set of factors determine the removal of development such as building condition and some factors may be inherently un-spatial.

4. In the development raster output from the UDM, cell values of two signify future development. To adapt this output for modelling decline, cell values of zero should be written to the output raster signifying cells made available which can then be used to update the rasters of constraint and current development for use by the next instance of UDM.

9.9 UDM Raster Data from Vector inputs

The first step in preparing data for the urban development model is to define the study region in terms of the attributes described by the asci raster header. The rasterised study region is buffered to avoid clipping perimeter data and the minimum and maximum extents are rounded to values which accommodate an exact quantity of cells using the specified cell size. The lower left corner values are the minimum extents of the rasterised study region in x and y respectively, whilst the number of columns and rows are the number of cells in the x and y directions respectively.

9.9.1 Standard Sampling

Input raster data for the study region is derived from vector layers describing various attributes of interest. These layers must be sampled and converted to the asci raster format before loading into the framework. To sample only the location of items in a vector layer it is practical to add a temporary integer column to the layer and set this to 1 for all rows, this will result in the production of a binary location raster when the subsequent steps are followed. The QGIS Regular Points function is used to create a vector layer which contains an array of points at a specified point spacing, covering a given range in x and y. Input coordinates are calculated from values in the asci raster header using the following equations:

$$X_{min} = X_{lower\ left\ corner} + \left({}^{cellsize}/_{2} \right)$$

$$X_{max} = X_{min} + \left((columns - 1) \times cellsize \right)$$

$$Y_{min} = Y_{lower\ left\ corner} + \left({}^{cellsize}/_{2} \right)$$

$$Y_{max} = Y_{min} + ((rows - 1) \times cellsize)$$

Where the offset $\left({}^{cellsize}/_{2}\right)$ is used to place each point at the centre of a corresponding raster cell. The point spacing is set to the cell size before creating and reimporting the resulting vector points layer.

This spatially uniform array of vector points is then used to sample data from the source vector layer. The QGIS Join Attributes by Location function is configured to keep all records including non-matching target records which results in a joined vector points layer that contains columns for all attributes from the source vector layer. These columns are populated with either valid data where an intersection was found or NULL values otherwise.

The joined vector points layer is converted to a raster via the QGIS Rasterize function, specifying the input shapefile, attribute field, and the raster resolution. This creates a raster of the desired resolution which contains either the value from the specified attribute field that was set to 1 earlier, or 0 for NULL values.

9.9.2 Super Sampling

Where reasonably small raster cells are used, it may be sufficient to use the standard sampling strategy to capture details of the source vector layer in the target raster. The method of using a single point sample at the centre of each raster cell becomes less reliable as the cell size increases and may lead to unwanted results. Consider a Boolean constraint raster used to mask development with a cell size of 100m, a cell which contains a small island in the middle of a lake, or a garden in an otherwise heavily developed area may be deemed suitable for development.

To mitigate this error a super sampling strategy can be employed using a linear scale factor which multiplies the number of cells in both x and y dimensions and divides the cell size. For example, using a linear scale factor of 10, a single 100m cell is replaced by 100 10m cells covering the same area which are sampled as before. These updated values for cell size, columns and rows are used to create a vector points layer with an increased sample density and the raster preparation proceeds as usual.

This super sampled raster can then be down sampled to restore the original values for cell size, columns and rows, and apply a threshold to the method of cell reduction. The linear scale factor is used to divide the number of cells in both x and y dimensions, and multiply the cell size, whilst the threshold determines how many input cells set to 1 in the super sampled raster constitute a cell set to 1 in the down sampled raster.

9.9.3 Zone Identity Raster

A raster of integer zone identity values must be created for the urban development model to correctly assign cells to zones. These values range from zero to the total number of zones minus one, and correspond to the index to the zone array used in the urban development model. Since zone order needs to be consistently defined across all models it is assumed that the zones vector layer has been sorted in ascending order using zone labels. A temporary integer column (id) must then be added to the zones layer which holds values incrementing in units from 1 to the total number of zones, a practical solution within QGIS is to set the values within this column equal to the row number. The standard sampling strategy can then be followed to produce a raster which holds values from the newly created (id) column, or 0 for NULL values which explains why values in the id column must start at 1.

Since zone identity values are to be used as an index to the zone array they must start at zero and increment in unit steps up to the number of zones minus one. To achieve this the 0 values in the raster which represent NULL i.e. non-zone values must be changed to some other no data value. A function is provided by the framework which loops through all raster cells and sets cells which are equal to a provided reference value (0 in this case) to the default no data value (-1). Valid zone identity values in the raster must then be adjusted so that the index starts at 0 instead of 1. Another function is provided by the framework which loops through all raster cells and adds a provided integer (-1 in this case) to all valid (i.e. not no data) cells.

9.9.4 Proximity Raster Data

The QGIS Proximity function can be used to calculate the shortest distance from each cell in a newly created target raster to a cell in the specified source raster with a given reference value (1 where the standard sampling strategy is used). Whilst a raster calculator is provided within QGIS which could be used to standardise the results, it was not found to be satisfactory so this functionality is provided by the framework. Two functions are provided which take proximity raster data as inputs and standardise the result using standard or reverse polarity using the following formulae:

$$X_{standard} = {(X - X_{min})}/{(X_{max} - X_{min})}$$

$$X_{reverse} = {(X_{max} - X)}/{(X_{max} - X_{min})}$$

Since the study region geometry is being mapped to a rectangular raster the maximum proximity value is often contained in a cell which does not belong to any zone. An optional input to each function is a mask, typically in the form of the zone identity raster. When this input is provided, cells with the no data value (-1) in the mask are not included in the standardisation equations. The proximity data is then standardised only for valid cells within the study region which leads to a better spread of standardised values across the 0 to 1 range.

9.10 UDM Downscaling

An experimental workflow was developed to bridge between land use rasters generated by UDM and models of urban drainage requiring higher spatial resolution. A method of converting average residential density at the census ward scale into a set of one-hectare tiles was developed by Hargreaves (2015) which enables the estimation of variability in residential land use and dwelling types. The specification of tile types includes density in dwellings per hectare (dph) and coverage percentages for permeable and impermeable surfaces. This work makes use of 16 of these tiles types, 4 in each housing category of Detached (D), Semi-detached (S), Terraced (T) and Flats (F) as shown in table 9.3.

Tile Type	**Buildings %**	**Green %**	**Roads and Paths %**	**Density (dph)**
D1	8	78	14	7
D2	13	73	14	12
D3	18	60	22	23
D4	20	56	24	30
S1	8	76	16	13
S2	13	68	19	23
S3	17	61	22	31
S4	20	50	30	42
T1	12	63	25	22
T2	29	39	32	68
T3	34	23	43	90
T4	34	15	51	109
F1	20	48	32	77
F2	21	41	38	101
F3	27	19	54	164
F4	27	27	46	216

Table 9.3: Tile Types, Ground Coverage and Densities adapted from Hargreaves (2015).

The specifications in table 9.3 were used to create a set of rasters to represent each of the 16 tile types as follows:

1. Create a one-hectare template using a cell size of 5m resulting in a raster region of 400 cells.
2. Convert the raster to a grayscale image.
3. Draw features for each surface category corresponding to the coverage percentage – ensure that roads connect across all tile types.
4. Convert the grayscale image back to raster.

This process was repeated for each row in table 9.3 to produce the set of 16 one-hectare tiles shown in figure 9.27

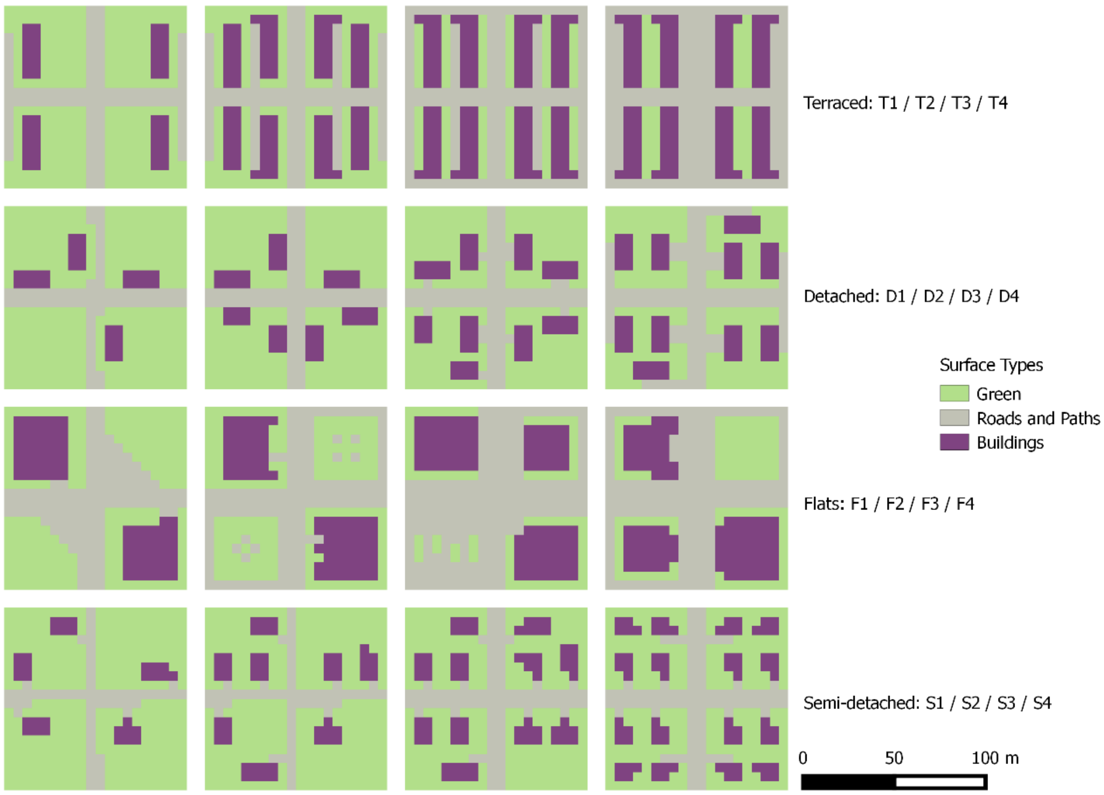

Figure 9.27: 1-hectare tiles with surface coverage percentages corresponding to table 9.3.

9.10.1 Downscaling Process

The tiles shown in figure 9.27 were then mapped to the UDM output as described in the following downscaling process:

1: Generate Gridded Network

It is assumed that drainage infrastructure for the test area follows the road network and is arranged in a simple grid. Network nodes are placed at the centre of each raster cell using regular points which are then connected by network edges.

2: Generate Density Surface and Bands

The test area follows a pattern of high density mixed-use central areas, through to medium-density residential through to low-density residential on the outskirts. The UDM uses spatial MCE to generate a raster of development suitability which in this case includes proximity to the point location at the centre of the test area. This proximity is used to generate a simple monocentric density surface which is then split into bands representing high, medium and low-density development.

3: Assign Tiles to Bands

The raster of density bands for the test area is used to generate a tile map where each band is populated with tiles randomly selected from the relevant set where tiles are grouped according to their number of dwellings per hectare (dph). The process of mapping raster cells from density bands to tile codes is shown in figure 9.28.

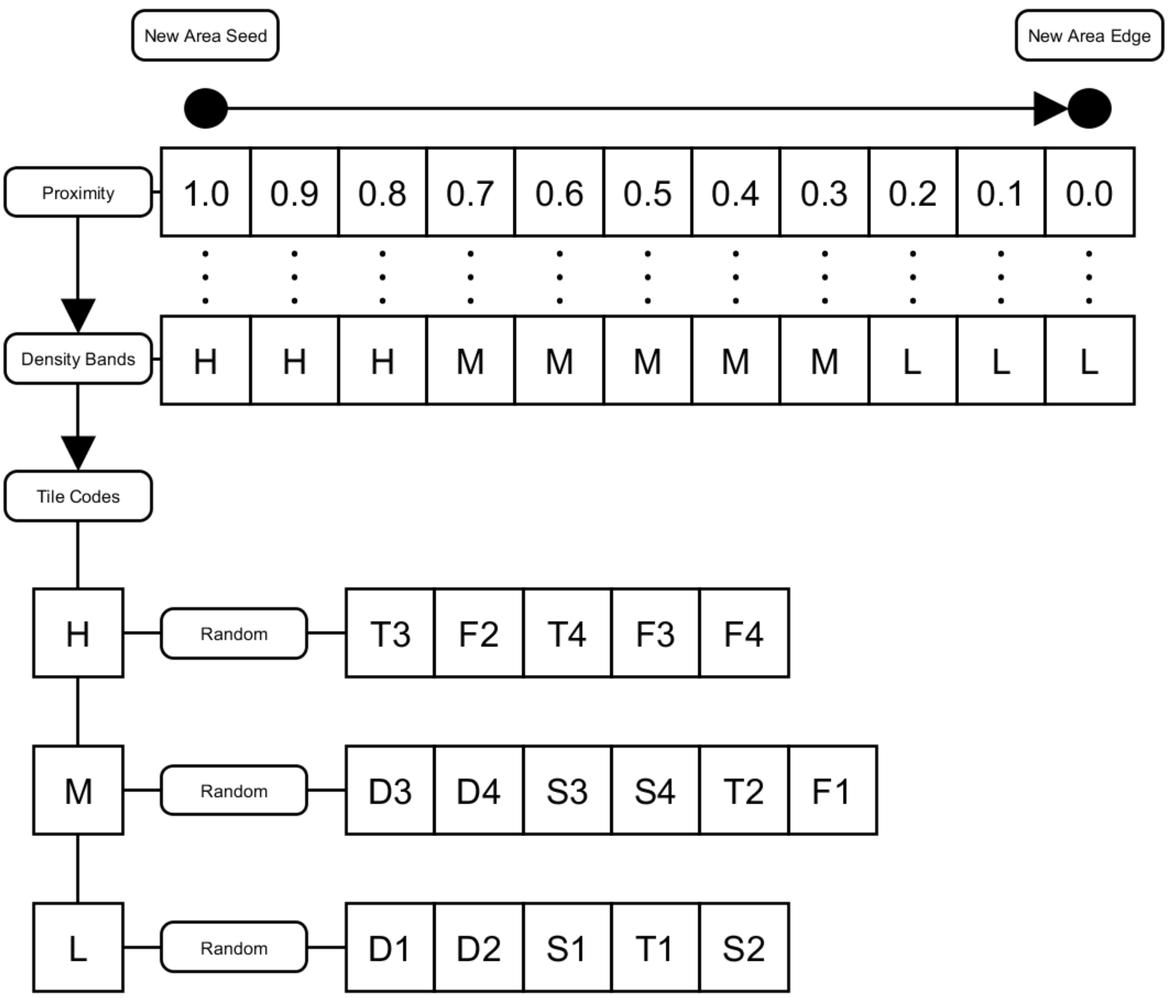

Figure 9.28: Overview of tile map generation process.

4: Generate Downscaled Development Raster

The tile map generated in the previous stage is used to generate a downscaled development raster for the test area including surface coverage types. This raster has the same extents in X and Y but has a cell size of 5m meaning that the number of columns and rows are multiplied by 20. Each cell in the tile map has an integer code referencing a 20*20 tile raster which is alternately rotated for variation and then mapped to the relevant subregion of the downscaled raster.

5: Split and Convert to Vector

The final step splits the downscaled development to generate a raster for each surface type which are then converted from raster to vector providing inputs to the urban drainage model along with the gridded network nodes and edges.

The five vector layers provided as input to the urban drainage model are shown in figure 9.29 which was draped over the topography of the test area using a 2m Digital Elevation Model (DEM). A design storm was then applied to the model demonstrating the ability to map flooded areas and estimate flood risk as shown in the resulting flood depth map shown in figure 9.30.

9.10.2 Downscaling Results

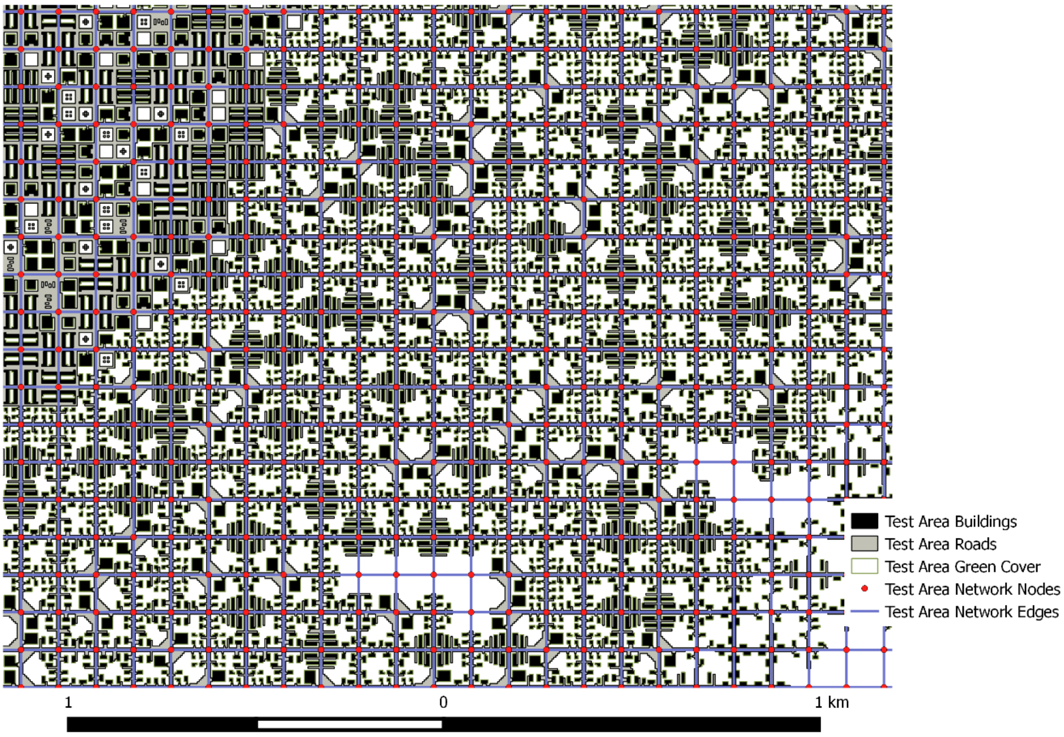

Figure 9.29: Gridded network and surface coverage variation driven by density from high (top-left) through medium to low (bottom-right).

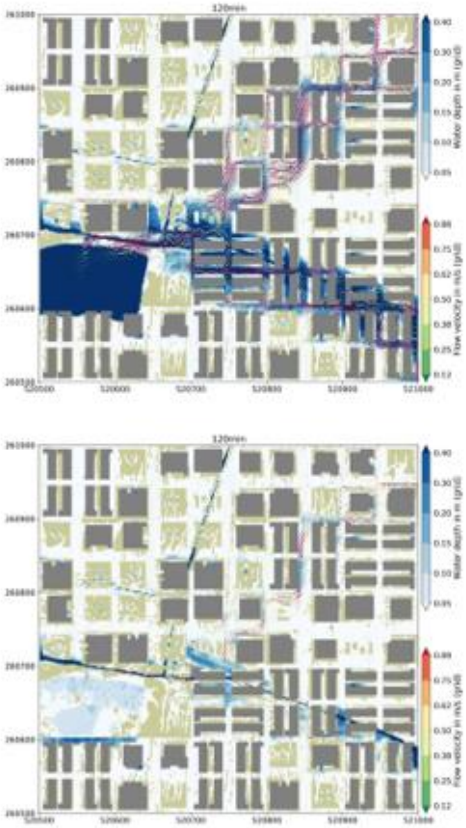

Figure 9.30: Flood depth maps for a 100-year design storm showing the effects of (top) inset roads, and (bottom) conventional drainage density.

9.11 Simplification

The primary function of the UIMF is to manage the integration and execution of models, providing the flexibility required to explore scenarios of land use and transport. The system of models developed in chapters 3 to 6 aim to provide a simple yet useful approach with minimum parameterisation requirements but crucially the architecture of the UIMF allows more complex approaches to be developed and integrated where necessary. Future work could expand upon existing capabilities in the areas described in the following subsections.

9.11.1 Spatial Interaction Model

1. Population age needed to simulate travel behaviour is not modelled by demographics.

   The SIM makes use of trips to work and therefore accessibility to employment to spatially disaggregate population; this is dominated by working age population which when driving the SIM with projected employment, is captured in the inverse activity ratio of employment and population. When driving the SIM with projected population, future population projections based on human fertility and migration do not include age; however, a population age profile could be provided by linking to a model such as SPENSER (Synthetic Population Estimation and Scenario Projection Model) being developed at the University of Leeds to give a full demographic breakdown of population.

2. Household income needed to simulate travel behaviour is not modelled.

   There is a lack of readily available data on income in the UK which could be used to drive such a model; however, modal split can be used to adjust the influence of accessibility via each mode of transport on resulting population patterns. In the UIAF Tyndall Cities project modal split for baseline scenarios was driven by empirical data as observed in the census data trip information. A minor adjustment is needed to incorporate this into the TAM so that trip costs via each mode are weighted to reflect modal split when combining cost matrices and converting to accessibility.

9.11.2 Transport Accessibility Model

1. Cycling and walking which are important for sustainable transport require further modelling.

Walking is included in the TAM by the assumption that intra-zonal costs are set at two-thirds of the minimum inter-zonal cost, and cycling is incorporated using a simple method of weighting network edges without monetary costs. Further considerations include the availability of additional network layers for walking and cycling and additional cost factors such as gradient and the perceived risk of cycling on roads with or without dedicated cycle lanes (Ford et al, 2015).

2. Traffic flows by vehicle type needed to model transport greenhouse gas emissions are not simulated.

   The UIAF and hence the UIMF is mainly targeted towards climate change adaptation rather than mitigation. In terms of traffic flows, the focus is on the land use patterns resulting from aggregate trips via all modes and vehicle types between origin and destination zones. Modal split can be used as an indication of shift between transport modes and associated vehicle types but transport GHG emissions can only be calculated using more complex methods.

3. Road congestion which is essential for the simulation of private vehicle traffic is not modelled.

   Congestion could be included as a measure of network capacity by adding costs to congested network edges to impede or redirect trips in response, this is similar to the congestion edge charging approach in the TAM. However, the incorporation of congestion as a deterrence measure across the network would require a more sophisticated approach than is currently used in the TAM such as microsimulation of traffic or an iterative equilibrium approach to determine which trips are affected.

9.12 UIMF Operation

9.12.1 Model Execution Times

The timings for running framework models in each case study (chapters 7 and 8) are repeated in tables 9.4 and 9.5. In both studies the TAM network processing models account for much of the running time of the framework which is why they are separated from other TAM stages; each network only needs to be (re)built to simulate physical changes such as the building of new roads or public transport stops. The most expensive network processing model in terms of time is for London's road network which has over 65k network edges and a congestion charge zone which requires additional geometry tests to model. All model stages other than network processing execute very quickly which supports multiple model runs to examine uncertainties and in terms of rapid assessment in a stakeholder dialogue setting the performance of the UIMF models is fit for purpose.

Model	**Execution time (seconds)**
TAM: bus network processing	297
TAM: heavy rail network processing	51
TAM: light rail network processing	38
TAM: road network processing	401
TAM: bus cost and pathfinding	10
TAM: heavy rail cost and pathfinding	4
TAM: light rail cost and pathfinding	4
TAM: road cost and pathfinding	23
TAM: accessibility	4
SIM	2
UDM	4

Table 9.4: GLA Case Study Model Execution Times

Model	**Execution time (seconds)**
TAM: network processing	45
TAM: cost and pathfinding	3
TAM: accessibility	1
SIM	2
UDM	6

Table 9.5: Innsbruck Case Study Model Execution Times

9.12.2 User Interface

The UIMF provides extensive functionality to configure and run groups of models and load datasets for model calibration; this workflow which is based on Python scripts is functional and highly flexible but should be improved in terms of usability by the development of a Graphical User Interface (GUI). In keeping with the open-source aim of this research, QGIS was used to prepare and visualise geographic data throughout this thesis and represents a logical choice for GUI development. QGIS is implemented using C++, Python and Qt, and extensive documentation is available including the $PyQGIS$ developer cookbook for plugins and scripting, the Python interface API and the C++ API (QGIS). In the first instance, this development should focus on allowing the user to parameterise and execute a single model from within QGIS using scripting to automate the generation of visual outputs. After applying this to all models, a GUI should be developed to allow the user to control the flexible configuration, iteration and execution of model groups provided by the UIMF. Further work is required to adapt and target this system for non-expert users who may not be comfortable using a GIS, this would involve the automation of spatial operations within QGIS and the development of a user-friendly GUI.

9.13 UIMF Assessment

The requirements of integrated modelling frameworks for stakeholder dialogue as described in section 3.3 are used to critique the UIMF in the following subsections.

9.13.1 Validity

The use of models coupled across spatial scales in the UIMF is capable of generating scenarios of future urban development for rapid assessment in terms of climate change mitigation and adaptation. Although relatively simple, the models are based on established theory which provides validity to model outputs which should be treated as plausible rather than predictive. The UIMF is transferrable and maintains validity when applied to new study regions by ensuring that all models (TAM, SIM, UDM) are parameterised consistently.

9.13.2 Transparency

The relatively simple models used in the UIMF can be conceptually described in terms which are easily communicable, making them suitable for interactive engagement with stakeholders and the public. The UIMF is implemented entirely using open source software with full source code transparency allowing further development and modification of the software architecture and individual models where required.

9.13.3 Usability

The UIMF provides a basic user interface via the minimal standardised model descriptions developed in the PMI (3.6.2). It is not necessary to modify any Python scripts in order to apply the framework to new study regions or scenarios, instead, this is achieved by editing template PMI tables for each model provided that input data has been prepared. This is still somewhat cumbersome so a graphical user interface (GUI) and tighter integration with GIS to prepare model input data would greatly improve the process.

The minimum set of data required by the UIMF is relatively modest, but the framework can make use of more input data where available, such as attractors in the SIM and UDM to represent additional spatial drivers of population and land use change. This is achieved by taking the number of attractors as a parameter and automating the process of loading and using named input datasets in the models themselves.

9.13.4 Flexibility

The UIMF simplifies the task of consistent configuration across models by formalising model interfaces using the PMI (3.6.2) and model coupling via model groups (3.6.3). For any given study region, each scenario run by the UIMF is specified simply as a particular combination of model inputs. Exploratory modelling requires that multiple scenarios are run to assess the implications of alternative policies. Uncertainties and sensitivities can be explored by executing the model multiple times whilst varying inputs. To facilitate this, the UIMF allows models to be grouped and executed in a range of configurations, providing control across iterations to adjust inputs, outputs and parameters (3.6.4). The results produced by any given model run can only be of lasting value if the means of their production in terms of the model and its data are preserved. The UIMF records the inputs, outputs and parameters of all model configurations in metadata tables (3.6.5).

9.13.5 Performance

The emphasis on rapid assessment suitable for stakeholder dialogue places a limit on model execution times. Python modules can be arranged and connected in a flexible manner to meet the requirements of a range of modelling scenarios. The performance of UIMF models has been optimised by interfacing Python and C++ models using SWIG, and by ensuring that slower operations are only performed where needed (3.6.6).

9.14 Summary

This chapter has described the utility of the UIMF developed in chapter 3 which is itself, the principal result of this research. In section 9.2 the framework has been shown to be transferable between study regions and the complexity of the parameterisation process is scalable allowing quick results to be generated which act as the basis for further model exploration. Section 9.3 demonstrated that the examination of model response for a range of possible scenarios is key to the assessment of future land use development. Section 9.4

examined the utility of the framework as applied to planning problems described in the case studies and GLA scenarios, identifying generic transferability and framework coherence as key properties. Section 9.5 showcased the flexible features of the generic modelling framework developed in section 3.3, describing the use of these techniques to explore densification scenarios. Section 9.6 described how framework flexibility, performance and metadata support the MC technique of uncertainty estimation. Section 9.7 showed that the UDM could be applied to both employment and population to give a more detailed representation of future land use development. Section 9.8 described how the UDM could be configured to model decline consistent with the approach in the SIM. The UIMF is the key contribution to knowledge of this thesis; the ability to adapt and apply the framework in a range of situations as described above demonstrates utility which far exceeds that of the original UIAF models which were further developed in the UIMF. Section 9.9 discussed the handling vector and raster data in the UIMF whilst section 9.10 addressed issues of spatial scale. Section 9.11 covered simplification in the models implemented within the UIMF whilst section 9.12 examined operational issues including model interfaces and execution times. Finally, section 9.13 critiqued the UIMF against the requirements for modelling with stakeholders identified in chapter 3.

The following chapter concludes the thesis and details the key findings and implications of this research.

Chapter 10. Conclusion

The aim of this research was to develop a flexible modelling framework to provide decision support to urban planners and stakeholders engaged in participatory modelling; by exploring the tensions and trade-offs of alternative spatial planning policy in scenarios of future land use and transportation, urban transitions can be made which account for climate change mitigation and adaptation measures.

The following objectives were set to meet this aim:

Objective 1: Review the field of urban planning in the context of climate change and sustainability to identify key drivers of spatial planning policy

Objective 2: Specify the modelling requirements to best support the decision-making processes identified in objective 1.

Objective 3: Review the field of urban modelling to identify and assess candidate modelling approaches.

Objective 4: Develop a modelling framework using techniques identified by objective 3, to provide decision support for planners and meet the requirements specified in objective 2.

Objective 5: Apply the modelling framework to study regions and modelling scenarios to demonstrate the utility of the approach.

Research and development carried out to meet these objectives is summarised in the following 5 subsections.

10.1 Review of Urban Planning in the Context of Climate Change and Sustainability

This objective was met by examining policy and spatial planning in the context of sustainable development including environmental considerations, establishing the need for advanced decision support tools to better assess the sustainability impacts of development.

Cities consume resources and generate greenhouse gas emissions disproportionately to their spatial extent, they are also especially vulnerable to disrupted resource supplies and the effects of climate change (McEvoy *et al*., 2012). Sustainability efforts are concentrated on high density populations in urban areas and the continuing process of urbanisation. In turn, urban sustainability is increasingly considered in the context of climate change (Carter, 2011).

The interaction between policies aimed at mitigation and those aimed at adaptation requires detailed planning to avoid positive interventions aimed at one strategy leading to negative consequences in another (Dawson, 2011).

In UK sustainability appraisals the consideration of climate change impacts and conflicting policies between development alternatives is highly limited and whilst the appraisals are broad in scope, they are subjective and lacking in quantitative detail. This element of the planning process has attracted criticism where a more detailed and analytical treatment of evidence and policy interaction is needed (Gibson, 2006). In addition, whilst the UK planning process requires the modelling of transport, there is no equivalent requirement for land use modelling to assess potential impacts. This lack of detail in the implementation seems to be contrary to the supposed shift in UK spatial planning from the use of sustainability as one factor in the consideration of competing land uses, to the stated principle aim of the planning system (DCLG, 2011).

Improvements to the conceptual design and deployment of models is needed to engage community with the requirements of mitigation and adaptation. Participatory approaches based on integrated modelling have great potential to explore solutions and understand the consequences of policy change. In the context of urban mitigation and adaptation policies, there is a need to convey the benefits and side-effects of proposals to the public (Ford et al., 2018).

10.2 Specification of Modelling Requirements for Decision-Support

The general requirements for decision-support in participatory modelling with stakeholders are as shown in table 10.1.

Requirement	Description
Validity	The ability of a model to reproduce changes in observed data should be demonstrated in model validation to minimise bias and provide confidence in model outputs
Transparency	Decision support tools aimed at resolving conflicting opinions on urban policy require transparency to be of use; opaque, or 'Black Box' models do not convey sufficient understanding to support argument. Transparency must be balanced with validity for a model to be of practical use i.e. simple models may provide greater transparency at the expense of reduced validity in representing the problem; whereas,

	complex models may provide a more valid problem representation at the expense of reduced transparency
Usability	To be deployed independently of developers, a model should be able to be configured, adapted and understood by model users and stakeholders. Usability should therefore be maximised whilst maintaining behavioural and empirical validity. Collating suitable data whilst accounting for errors and omissions, and ensuring consistency poses a significant task which is not helped greatly by currently available software. Exhaustive input requirements and the complexity of data development can deter users and prevent model implementation.
Flexibility	To be applicable to a wide range of users and purposes a model system must have enough flexibility to handle variations in input data and model output requirements. The development of decision support tools must also respond to advances in data, software and theory to provide new solutions.
Performance	Computational performance must be considered alongside the other challenges of integrated modelling; for instance, poor performance can restrict the use of otherwise viable models whereas models which aim for interactive runtimes can compromise validity.

Table 10.1: Framework Requirements – adapted from Waddell (2011).

10.3 Review of Urban Modelling Techniques

This objective was met by reviewing urban modelling theory and identifying techniques to develop an appropriate modelling and assessment system.

The overwhelming trend in urban modelling is the disaggregation towards microsimulation; this is driven by the development of more powerful computers and the availability of disaggregate data from GIS-based applications. These resources are coupled with complex systems approaches using CA and ABM, and models of both land use and transport using microsimulation. Modelling at the individual level has the benefit of improved conceptual theories of behaviour, interactions and preferences which influence mobility and patterns of location; however, this must be balanced against the associated practical drawbacks of large data requirements and long computing times. Microsimulation models are also prone to a lack of stability arising from stochastic variation when different random seeds are used across model iterations; this can obscure the response to variations of model inputs preventing spatial policy assessment (Wegener, 2014).

The dominant approach to urban modelling aims to predict the future state of the urban system with some degree of accuracy using complex models; contrasting this is the use of simpler models whose objective is not to predict, but to explore the parameters and scenario space across multiple model iterations to inform discussion (Batty, 2013). There is a scarcity of simple urban models which are suited to the task of integrating land use and transportation with the environment; in many cases the data requirements, difficulty of calibration and slow execution times prevent the adoption of existing models for this purpose (Mikovits *et al*., 2014).

In terms of the complexity of urban land use and transport models, each planning problem has bespoke model requirements regarding scale and complexity when the practical tasks of data gathering, calibration and model execution time are considered; the optimum model is one which outputs just enough detail to address the problem whilst minimising these practical costs. “Future urban models will be modular and multi-level in scope, space and time.” (Wegener, 2011, pp 171)

10.4 Development of Modelling Framework

The aim of this research was to develop a simple, transferable and consistent modelling system suited to the purpose of generating plausible scenarios of future development for analysis along with environmental factors. Building such a system from first principles in a relatively short timeframe was not feasible so an existing framework was identified which has desirable properties but does not meet all of the requirements of integrated modelling frameworks for stakeholder dialogue as summarised in table 10.1.

The UIAF was developed in a modular fashion by separate teams using their own preferred software and languages, these modules are integrated in a process described at best as loose coupling to simulate transport, locational choice and land use in the Greater London area. The requirements summarised in table 10.1 were used to critique the UIAF to define design criteria for an integrated modelling framework. The methodology developed by this research unifies the multi-scale LUTM components of the UIAF in an integrated modelling framework in which the LUTM scenario generation system is separated from the environmental impact models. This collection of models along with their implementation in the specific software environment developed by this research constitutes the Urban Integrated Modelling Framework (UIMF).

As the UIMF is the principle output of this research the key features are summarised as follows:

10.4.1 Generic Models and Data

The framework standardises model configuration and execution via a python model interface; all models are executed from Python scripts linked to Postgres tables which hold model datasets. Three modes of model group iteration are supported allowing models in the group to be executed once, n-times or until a stop condition reported by a model within the group is reached.

10.4.2 Metadata

To provide traceability, the results produced by framework models are accompanied by metadata which documents the model used and its datasets. Models running under the framework can be arranged in model chains where the output of one model serves as the input to another; the handling of metadata in the framework mirrors this by stitching together metadata into chains.

10.4.2 Model Drivers

In many cases when iterating over model groups, the model inputs, outputs and parameters must be changed for each group iteration. The framework abstracts this flow control and routing from the models themselves using model drivers which promotes model reusability, allowing models to be arranged and rearranged without modification of their Python scripts.

10.4.3 Model Group Lists

Framework models, model drivers and metadata can be chained together in model groups; this functionality is extended using model group lists to accommodate scenarios where multiple model groups are required in which each model group performs a given task within the overall simulation. Using these foundations, the user can build model group chains where each model group has its own iteration control and model driver to control data routing.

10.4.4 Computational Flexibility

Models are executed via Python scripts using embedded SQL commands to interact with data stored in Postgres spatial database tables; the interactivity and flexibility of Python allows models to be connected in a flexible manner for a range of modelling purposes. To improve performance whilst retaining flexibility, the framework uses SWIG to create wrappers which interface C/C++ code with Python.

10.4.5 Data Handling

The framework provides numerous functions and techniques to manage the data transfer process involving csv, vector, and raster datatypes.

10.4.6 Spatial Interaction Model (SIM)

Projected values of employment and population are converted to ratios which scale the respective observed value allowing the change in employment and population within the study region to be driven in a flexible manner. Modifying the available area input allows a range of constraints to be applied reflecting different land use development policies. Employment is spatially disaggregated using normalised weighted attractors reflecting development policy for the study region. The spatial disaggregation of population uses normalised weighted attractors to reflect development policy, along with accessibility to employment via the matrix output from the transport model.

10.4.7 Transport Accessibility Model (TAM)

The private network processing model connects zone centroids to the input transport network and optionally performs spatial checks in support of a flexible charging scheme. The public network processing model connects zone centroids to one or more stations and creates interchange links between stations within a user-defined radius of one another. Network cost models use input cost parameters to calculate network edge weights used in the computation of shortest paths; an origin-destination cost matrix is generated which, for public networks, accounts for all possible routes for zone centroids connected to multiple stations. The accessibility model aggregates all input cost matrices using the log sum of exponentials.

10.4.8 Urban Development Model (UDM)

The first UDM stage employs multi criteria evaluation to produce a raster of development suitability from weighted suitability inputs and development constraints. The second UDM stage groups cells into development areas within each zone of a specified minimum size, before computing the average development suitability for each area using the raster output from multi criteria evaluation. The final UDM stage develops cells within development areas in each zone according to zonal population change and zonal population density using the suitability for both development areas, and individual cells to distribute development.

10.5 Framework Applications

In chapters 7 and 8 the UIMF was parameterised in case studies for the GLA region, a major global city, and for the city of Innsbruck and its surrounding region; the Inn Valley, in the western Austrian state of Tyrol. The GLA region was parameterised using detailed spatial planning policy, multiple modes of transport and generalised cost parameterisation whereas the Innsbruck study region used a single spatial policy attractor, a single transport network and distance-based cost; this demonstrates that the UIMF can make use of detailed parameterisation data, where available, and can also be parameterised using a sparse set of data.

A more comprehensive demonstration of the application of the UIMF is presented in Chapter 9.3 where a suite of scenarios combining spatial policy drivers, transport infrastructure investment and climate impacts is used to drive the UDM. Four land use scenarios are used representing extreme policy drivers, these are simulated using UIMF models to produce spatial outputs of land use, population and density for the year 2100 which permits the comparative assessment of flood risk across all scenarios.

10.6 UIMF Assessment

The requirements of integrated modelling frameworks for stakeholder dialogue as described in section 3.3 are used to critique the UIMF in the following subsections.

10.6.1 Validity

The use of models coupled across spatial scales in the UIMF is capable of generating scenarios of future urban development for rapid assessment in terms of climate change mitigation and adaptation. Although relatively simple, the models are based on established theory which provides validity to model outputs which should be treated as plausible rather than predictive. The UIMF is transferrable and maintains validity when applied to new study regions by ensuring that all models (TAM, SIM, UDM) are parameterised consistently.

10.6.2 Transparency

The relatively simple models used in the UIMF can be conceptually described in terms which are easily communicable, making them suitable for interactive engagement with stakeholders and the public. The UIMF is implemented entirely using open source software with full source code transparency allowing further development and modification of the software architecture and individual models where required.

10.6.3 Usability

The UIMF provides a basic user interface via the minimal standardised model descriptions developed in the PMI (3.6.2). It is not necessary to modify any Python scripts in order to apply the framework to new study regions or scenarios, instead, this is achieved by editing template PMI tables for each model provided that input data has been prepared. This is still somewhat cumbersome so a graphical user interface (GUI) and tighter integration with GIS to prepare model input data would greatly improve the process.

The minimum set of data required by the UIMF is relatively modest, but the framework can make use of more input data where available, such as attractors in the SIM and UDM to represent additional spatial drivers of population and land use change. This is achieved by taking the number of attractors as a parameter and automating the process of loading and using named input datasets in the models themselves.

10.6.4 Flexibility

The UIMF simplifies the task of consistent configuration across models by formalising model interfaces using the PMI (3.6.2) and model coupling via model groups (3.6.3). For any given study region, each scenario run by the UIMF is specified simply as a particular combination of model inputs. Exploratory modelling requires that multiple scenarios are run to assess the implications of alternative policies. Uncertainties and sensitivities can be explored by executing the model multiple times whilst varying inputs. To facilitate this, the UIMF allows models to be grouped and executed in a range of configurations, providing control across iterations to adjust inputs, outputs and parameters (3.6.4). The results produced by any given model run can only be of lasting value if the means of their production in terms of the model and its data are preserved. The UIMF records the inputs, outputs and parameters of all model configurations in metadata tables (3.6.5).

10.6.5 Performance

The emphasis on rapid assessment suitable for stakeholder dialogue places a limit on model execution times. Python modules can be arranged and connected in a flexible manner to meet the requirements of a range of modelling scenarios. The performance of UIMF models has been optimised by interfacing Python and C++ models using SWIG, and by ensuring that slower operations are only performed where needed (3.6.6).

10.7 Key Findings and Implications

10.7.1 Key Findings

1. That it is possible via the framework developed by this research to implement and undertake robust scenario evaluation of climate impacts on cities within an integrated modelling framework. This flexibility is provided by a standardised model interface and the controlled iteration and execution of chained groups of models linked to PostgreSQL tables holding model datasets and metadata.
2. That relatively simple separate models can be coupled in such a manner that multi-scale planning options can be investigated to provide information and insights that are broadly comparable with more complex models which require more input data and are more difficult to parameterise.
3. That results for the GLA scenarios described in Section 9.3 demonstrate the potential impact of spatial policy in terms of the area and population density of new development in the floodplain which increases risk. This climate threat is factored into a range of alternative policies including the relaxation of the greenbelt constraint and the inclusion of a flood constraint which result in a further set of model outputs representing the competing pressures faced by urban planners.
4. That the Innsbruck case study presented in Chapter 8 demonstrates the framework can operate using minimum data for simple study region parameterisation. The main objectives for the case study are achieved by demonstrating the transferability and usability of the framework while the results generated by simple model parameterisation form the basis for further work exploring policy scenarios and potential impacts for the Innsbruck region.

10.7.2 Implications

1. The demonstrated generic transferability of the framework permits the investigation of different regions, planning contexts and applications. The coherency provided by the framework in terms of model parameterisation and execution is essential to its utility and flexibility as a decision-support tool.
2. The modular nature of the framework facilitates the inclusion and integration of new models as they are developed. The examination of rapid transitions in policy aimed at climate change and sustainability requires the integration of models from many disciplines involving different modelling paradigms; the development of tools for

decision support must respond to advances in data, software and theory to provide new solutions.

10.8 Further Framework Applications

Section 9.5 considers two alternative methods of exploring densification scenarios involving the density adjustment of targeted and non-targeted zones to accommodate projected future development. In each method, two driver models and model group iteration are used to automate and control the densification scenarios. The first driver model controls the density adjustment process providing modified values which are input to the zonal model, while the second driver model stops group iteration when employment and population can be accommodated for all timesteps. A range of different target zones and associated densities could be tested to generate a portfolio of options to inform densification policy.

Section 9.6 describes a four-step process to estimate model uncertainty which could be automated using model drivers and model group iteration to modify model inputs, execute the model, and store model outputs, along with metadata for all required iterations. The time taken to compute the uncertainty estimation depends on the execution time of the model under test; the performance of the UIMF minimises this computational cost and permits the generation of output values from a large number of model executions in a short timeframe.

Section 9.7 describes the configuration of the UIMF using multiple UDMs to map the change in both employment and population over all simulated timesteps. Model drivers and model group iteration are used to control and coordinate the multiple UDMs used in the modelling arrangement which involves the update of raster data between UDM runs to reflect changes in development along with model feedback across iterations to synchronise UDM land use change for employment and population development with values generated by the zonal model. This permits a more detailed exploration of the temporal and spatial distribution of land use change attributed to development for both employment and population which could be used to examine the localised effects of surface sealing and urban drainage network modification.

The application of most urban models is restricted to scenarios of urban expansion and the associated land use demands of future development; however, the UIMF SIM can also spatially disaggregate employment and population where the change in either value for the current timestep is negative which results in land being given back to the zone in accordance with the zone’s specified development density. The UDM considers positive population change only, so to address this inconsistency across modelling scales, Section 9.8 describes how the UDM could be used to output a fine-scale representation of land reuse, repurposing or demolition for zones assigned negative change by the zonal model.

Climate impact models overlay the spatial footprints of climate hazards with the patterns of land use generated by the UIMF to spatially assess vulnerability and exposure. The potential impacts of tidal and fluvial flooding were discussed in the GLA scenarios in terms of the interaction with future population development outputs. The outputs generated by the UIMF could be used in conjunction with a range of environmental models to assess the impact of multiple climate hazards including the use of future extremes of precipitation to assess pluvial flooding and surface-water flows, and temperature simulations to assess air quality attributed to emissions and excessive heat. Section 9.9 describes an experimental downscaling workflow which was developed to bridge between land use rasters generated by UDM and models of urban drainage requiring higher spatial resolution.

Climate impacts on infrastructure can be direct such as the flooding of buildings, or indirect such as the disruption to transport networks caused by flooding and temperature extremes negatively effecting performance in terms of capacity and journey times. Future climate extremes of temperature and the associated reduction in air quality also pose serious health threats to urban citizens. There is a need to assess adaptation options and test their effectiveness against all potential impacts to provide a comprehensive basis from which to explore tensions between mitigation and adaptation, for example, densification to reduce transport related greenhouse gas (GHG) emissions Vs more intense urban heat island effect.

The UIMF could be applied to more general consideration of future urban liveability to construct and analyse a suite of policies aimed at the development of sustainable urban areas; this would include sustainable transport goals such as reducing emissions, promoting low-carbon modes of transport and reducing long-distance commuting. Options to improve urban mobility and minimise congestion could be explored along with an expanded model of accessibility considering access to employment as well as a range of other locations such as retail and green space.

10.9 Further Research

10.9.1 Advanced Analysis

The use of C++ to implement relatively simple models in the UIMF results in an application capable of generating near-real-time results for scenario exploration in an interactive stakeholder dialogue setting; however, the communication to stakeholders of model uncertainties and sensitivities can require the generation of many thousands of model results involving a far greater computational cost. Further research is required to develop suitable UIMF performance for this task; this should be accompanied by the development of analytical tools based upon software in the SciPy Stack for instance the Python Data Analysis Library (pandas) and Matplotlib.

One option for accelerating the UIMF codebase on systems with the requisite hardware is the use of general-purpose computing on graphics processing units (GPGPU). The massively parallel architecture of graphics cards can be harnessed using CUDA or OpenCL to achieve significant performance gains, the former being preferable in terms of documentation, development tools and supporting libraries. Custom routines can be developed to perform computation on the GPU taking charge of the fundamental steps of writing host code on the CPU to interface with the device and manage memory, and kernel code to implement computation which executes on the GPU. Another option is the use of distributed computing; Condor is a management system supporting high throughput computing (HTC) for dedicated compute nodes which is well suited to tasks which apply the same processing to many different datasets. Condor jobs are submitted via a command file, queued, and executed on available computers, making use of configuration files or the network to communicate. Given that PC clusters are highly heterogeneous computing environments care must be taken to ensure that executable code has the best chance of running on any given machine; this can be achieved by statically rather than dynamically linking any code dependencies.

The UIMF supports the multi-scale analysis of policy, land use, transport and climate impacts by generating future patterns of development over long timeframes; further research and improvements are required to model processes of rapid change and sub-cellular land use to provide greater insights into the detailed spatiotemporal pathways to improved adaptation and planning.

10.9.2 Advanced Visualisation

QGIS provides a fit-for-purpose solution for visualising model inputs and outputs in single model runs as presented in Chapter 7; however, more advanced use of the UIMF such as the generation of results for uncertainty analysis and parameter sweeping across multiple models runs can generate a huge number of results. The problem posed in this situation is how to visualise the outcomes from multiple model runs in way which can be easily interpreted to aid the model assessment process. The QGIS rendering system is not designed for high throughput, does not employ hardware acceleration using a Graphics Processing Unit (GPU) and provides only a fixed set of visualisation capabilities which severely limits its application to the task at hand. OpenGL is an API for hardware-accelerated graphics used in video games and scientific visualisation. Since version 2.0 was released in 2004, the API includes a C-like language which can be used to create bespoke shading programs which are executed on GPUs exploiting the vastly superior performance in comparison with CPUs. The current OpenGL version is 4.6 and the API includes a vast array of extensions which can be applied to the visualisation of complex data. A key feature is the buffering of generic data from the CPU which is linked to objects rendered on the GPU; in conjunction with the speed of geometry throughput and the rendering flexibility offered by shaders, this provides an adaptable and high-performance solution to the problem of visualising large datasets. Further research in this area is required to develop a visualisation system capable of communicating the results from multiple model runs in a timely and comprehensible manner.

10.9.3 User Interface

The UIMF provides extensive functionality to configure and run groups of models and load datasets for model calibration; this workflow which is based on Python scripts is functional and highly flexible but should be improved in terms of usability by the development of a Graphical User Interface (GUI). In keeping with the open-source aim of this research, QGIS was used to prepare and visualise geographic data throughout this thesis and represents a logical choice for GUI development. QGIS is implemented using C++, Python and Qt, and extensive documentation is available including the *PyQGIS* developer cookbook for plugins and scripting, the Python interface API and the C++ API (QGIS). In the first instance, this development should focus on allowing the user to parameterise and execute a single model from within QGIS using scripting to automate the generation of visual outputs. After applying this to all models, a GUI should be developed to allow the user to control the flexible configuration, iteration and execution of model groups provided by the UIMF. Further work is required to adapt and target this system for non-expert users who may not be comfortable using a GIS, this would involve the automation of spatial operations within QGIS and the development of a user-friendly GUI.

10.10 Contribution to Knowledge

In summary, this research contributes to the field of spatial analytics and modelling in the following key areas:

1. The reimplementation of existing UIAF models using open source software and coding platforms.

   The UIAF models were all implemented using proprietary software and in some cases are no longer supported due to licencing issues – this work ensures that those models can be used and developed in further research.

2. The development of a generic and flexible framework in which spatial models can be specified, integrated and executed.

   The UIMF reimplements the LUTM components of the UIAF but it could be used in a variety of situations where spatial (and non-spatial) models require integration.

3. The implementation of the UIMF which can be used for rapid assessment in a stakeholder dialogue setting and also for more robust analysis involving uncertainties and sensitivities which require multiple model runs coupled with metadata to record the provenance of results.

Appendix A. Datatypes and Data Flows

The management and transfer of data between Postgres database tables and models which consume and produce that data should be handled within the Python module responsible for executing each model. The UIMF provides numerous functions and techniques to assist in this data handling process which involves csv, vector, and raster data types. The discussion of these techniques begins with csv data in the context of the user setting up generic datasets required by each model to be run using the PMI. These datasets could be setup within PostgreSQL using the $pgAdmin$ interface; however, doing so is somewhat cumbersome which lead to an alternative method being used in which the generic datasets are created and edited externally and uploaded to the database via csv files.

Generic model inputs and outputs share the same table format with columns for primary key, name and value being completed for each item of data. The first step in managing this data via an external source is to create a named database table of the required format. SQL code for table creation can be uploaded to the database and called by Python functions with the table name as an argument; this is an appropriate solution for table formats which are regularly reused as is the case here. The primary purpose of this SQL code is to $EXECUTE$ the $CREATE\ TABLE$ instruction using the name provided before defining the name and type of each column to be included in the table.

The only table format in section which is not fixed is that for parameter tables which have columns for primary key, model key and a column for each parameter assigned to the model. Parameter tables can be created as described previously where each table definition requires a separate function, an alternative solution is to further generalise the process of table creation; a generic function $create\_empty\_table()$ can be used to call the SQL commands $EXECUTE$ and $CREATE\ TABLE$ along with the table name provided as an argument in the Python function to create a table with no columns. The SQL commands $ALTER\ TABLE$ and $ADD\ COLUMN$ can then be used to format the generic parameter table as required.

After a table is created, it can be populated with data from a suitably formatted csv file from within a Python script using a generic $data\_to\_table()$ function which uses the SQL commands $COPY, FROM$, $DELIMITERS$ and $CSV\ HEADER$ along with the names of the database table and input csv file to upload the data. Note that the csv file must be publicly accessible for Postgres to execute the data transfer.

Inputs named in generic PMI model datasets can refer to vector data providing the database used is spatially enabled using the $PostGIS$ extension. Vector data can be uploaded directly within $pgAdmin$ using the shapefile import/export manager plugin or can be handled in a similar manner to csv data by creating the table and loading from a suitably formatted csv file; a generic $data\_to\_geom\_table()$ function extends the aforementioned $data\_to\_table()$ function by enforcing the provided spatial reference and dimensions of the geometry column.

The techniques described in this section so far allow the user to upload data to specify generic model datasets and input data; these methods are also utilised during model execution to write outputs to named database tables while further techniques are provided for models to read inputs from database tables. A set of functions prefixed $extract\_$ provides the ability for models to pull data from tables. In cases where the entire table is required and data is suitably ordered, $extract\_all()$ uses the SQL commands $COPY, SELECT, FROM, TO$ and $CSV\ HEADER$ along with the name of the database table and output csv file to export the data. Note again that the csv file must be publicly accessible for PostgreSQL to execute the data transfer. Where only a subset of the table is required, the SQL command AS is used in the $SELECT$ statement along with the provided name for each column to be exported. Where data order needs to be specified, the SQL command $ORDER\ BY$ can be used to arrange the rows of exported data according to the row order of a named column while $DESC$ can be used to order the exported rows in reverse.

The use of ascii raster files provides a simple format to exchange raster data along with the following header information:

- $ncols$ (number of columns)
- $nrows$ (number of rows)
- $xllcorner$ (lower-left-corner, lower-left-cell, x coordinate)
- $yllcorner$ (lower-left-corner, lower-left-cell, y coordinate)
- $cellsize$ (size of each cell)
- $NODATA\_value$ (cell data mask value)

The raster data follows the header starting at the top left i.e. the first row and first column of the raster. For each row, the cell value in each column is recorded using whitespace delimiters before moving to the next row to complete all cells in the raster.

Raster data stored in the ascii format can be handled within C++ models using a simple Raster class containing member variables for each item of header information; the $ncols$ and $nrows$

header items are used to dynamically allocate a 2d array into which cell values are loaded using $stringstream$ and $string$ types to read data from file and convert to $integer$ or $double$ data types as required.

The ascii format is used as the UIMF raster interface in terms of importing and exporting data which can be prepared and viewed in $QGIS$ but since a large amount of raster data may need to be transferred between models and to/from Postgres some modifications have been made to improve performance. The $PostGIS$ raster type (formerly $PostGIS\ WKT\ Raster$) was first included in $PostGIS$ version 2.0 permitting the storage and querying of raster data stored in Postgres tables. All UIMF models to date load and save rasters via local files but raster data must be stored alongside other model data and metadata to fully record any given model run. Since the $PostGIS$ raster type was not required it was decided to store raster data in standard Postgres tables using a single column to standardise table creation, and to seek to improve data transfer speed when working with ascii files. To preserve the use of standard ascii files for raster import and export the C++ Raster class was modified to use a setup function which takes a header file ($.hdr$) containing the 6 items from the ascii file format header and dynamically allocates storage as before. The Raster class was then expanded with functions to load and save data using two alternative formats; single column csv and binary.

The C++ code to read raster data from a single column csv file reads the entire raster into a one dimensional $string$ array before converting to $integer$ or $double$ data types and re-indexing as required. This provides a considerable increase in performance when compared with reading from ascii files in which each line of data representing a raster row is read from file into a $string$, then into and out of a $stringstream$ to skip whitespace before conversion. The methods to transfer this data to and from a single column database table are as described previously in this section for importing and exporting csv data.

Reading an entire raster from a single column csv file using the basic csv interface developed for the UIMF still involves reading data for each line; a better strategy for fast data transfer would be to handle the entire raster using a single read operation. Binary data in raw form would be easy to manage e.g. for rasters containing integer values the data would be the number of cells multiplied by $sizeof(int)$ in C++ parlance. In practice, binary data transfer to and from Postgres is faster (byte for byte) than other formats but involves considerable data redundancy and coding effort to manage the transfer; reading and writing binary data to database tables is simply a matter of including a format option wrapped within the generic function $binary\_data\_to\_table()$.

The SQL $COPY$ command has a parameter $FORMAT$ which specifies the type of data to be transferred. One of these formats is $binary$ which offers improved data transfer performance at the expense of machine architecture portability and data type flexibility. The binary format includes a file header, padded tuples of row data and a file footer, all of which are in network byte order.

The problems of handling binary data as used by Postgres lie in the data padding used which greatly increases file sizes, and the network byte ordering (endianness) which is counter to that used on most operating systems. Reading raster data involves a 19-byte header and a 2-byte footer, then each cell value (4-bytes for integer, 8-bytes for double) is preceded by 6-bytes of padding which bloats the stored raster file. The goal of reading the entire raster in a single operation is achieved but converting endianness from most significant byte to least significant byte or vice versa results in the use of staggered reverse and copy operations to manipulate data. The binary method of transferring raster data is the still the fastest option but is not necessarily portable since the code relies on $sizeof(datatype)$ which may vary across machine architectures and assumes that the endianness of the client is different from the server. Using an alternative raster format within model code could avoid many of the issues encountered and provide a means of quickly transferring raster data between models and to/from the database; however, the methods developed address performance problems associated with the ascii format to provide a working solution fit for purpose.

Both the single column csv and binary raster exchange formats benefit significantly from the use of unlogged tables to transfer data. The SQL command $CREATE\ TABLE$ has a parameter $UNLOGGED$ which specifies whether table data is written to the write-ahead-log confirming data integrity. The use of unlogged tables significantly improves the performance of data transfer; however, this performance increase is traded against the loss of protection in the event of a system crash since the table contents in the secondary server used by the write-ahead-log will be empty when disabled.

References

Acheampong, R.A. and Silva, E. (2015) 'Land use–transport interaction modeling: A review of the literature and future research directions', *The Journal of Transport and Land Use*, 2015 8(3), pp. 11-38.

Alonso, W. (1964) *Location and Land Use: Toward a General Theory of Land Rent*. Cambridge, Mass: Harvard University Press.

Anas, A. (1984) 'Discrete Choice Theory and the General Equilibrium of Employment, Housing, and Travel Networks in a Lowry-Type Model of the Urban Economy', *Environment and Planning A*, 1984 16(11), pp. 1489-1502.

Bai, X, Dawson, R, J, Ürge-Vorsatz, D, Delgado, G, C, Barau, A, S, Dhakal, S, Dodman, D, Leonardsen, L, Masson-Delmotte, V, Roberts, D, C and Schultz, S. (2018) 'Six Research Priorities for Cities and Climate Change.', *Nature,* (2018) 555: pp. 23-25.

Barra, T. de (1989) 'Applications of TRANUS, an integrated land use and transport model', in Barra, T. *Integrated Land Use and Transport Modelling*. Cambridge: Cambridge University Press. pp. 143–167.

Batty, M., (2000). 'Geocomputation using cellular automata' in Openshaw, S and Abrahart, R, J, *GeoComputation*. New York: Taylor & Francis. pp. 95-126.

Batty, M. (2007) *Cities and Complexity: Understanding Cities with Cellular Automata.* Cambridge, MA: The MIT Press.

Batty, M. (2007b) 'Spatial Interaction', in Kemp, K. *Encyclopedia of Geographic Information Science.* Thousand Oaks, CA: SAGE, 2007. pp. 417-419.

Batty, M. (2013) 'Visually-Driven Urban Simulation: Exploring Fast and Slow Change in Residential Location', *Environment and Planning A*, 2013 45(3), pp. 532-552.

Batty, M., Vargas, C., Smith, D., Serras, J., Reades, J. and Johansson, A. (2013) 'SIMULACRA: Fast Land-Use—Transportation Models for the Rapid Assessment of Urban Futures', *Environment and Planning B: Planning and Design*, 2013 40(6), pp. 987-1002.

Berkhout, F., J. Hertin, and A. Jordan. (2002) 'Socio-economic Futures in Climate Change Impact Assessment: Using Scenarios as 'learning Machines'. *Global Environmental Change,* 2002, pp. 83-95.

Bollinger, L. Andrew, and Ralph Evins. (2015) 'Facilitating Model Reuse and Integration in an Urban Energy Simulation Platform'. *Procedia Computer Science*, 2015 51, pp. 2127-136.

Brundtland, G, H. (1987). *Report of the World Commission on Environment and Development: Our Common Future*, Oslo, Norway: Available at: https://www.are.admin.ch/are/en/home/sustainable-development/international-cooperation/2030agenda/un-_-milestones-in-sustainable-development/1987--brundtland-report.html (Accessed: 07/11/2020).

Bulatewicz, T., Allen, A., Peterson, J, M., Staggenborg, S., Welch, S, M., and Steward. D, R. (2013) 'The Simple Script Wrapper for OpenMI: Enabling Interdisciplinary Modeling Studies'. *Environmental Modelling and Software*. 2013 39, pp. 283-94.

Carter, J.G., (2011) 'Climate change adaptation in European Cities'. *Current Opinion in Environmental Sustainability*, 2011 3, pp.193–198.

Cervero, R., Ferrell, C. & Murphy, S. (2002) 'Transit-Oriented Development and Joint Development in the United States: A Literature Review'. *Research Results Digest*, 2002 52, pp.1–144.

Chapin, F, S. (1974) *Human activity patterns in the city; things people do in time and in space*. New York: New York, Wiley.

Chaudhuri, G. and Clarke, K. (2013) 'The SLEUTH land use change model: A review', *Environmental Resources Research*, 2013 1(1), pp. 88-105.

Clarke, K.C. (2014) 'Cellular automata and agent-based models.' In Fischer, M, and Nijkamp, P. *Handbook of Regional Science.* Berlin: Springer-Verlag. pp. 1217-1233.

Crosetto, M., Tarantola, S. and Saltelli, A. (2000) 'Sensitivity and uncertainty analysis in spatial modelling based on GIS', *Agriculture, Ecosystems and Environment*, 2000 81, pp. 71-79.

C40 Cities (2018a): *C40 Climate Action Planning Programme.* London: C40 Cities Leadership Group. Available at: www.resourcecentre.c40.org (Accessed 07/11/2020).

C40 Cities (2018b) *27 C40 CITIES HAVE PEAKED THEIR GREENHOUSE GAS EMISSION.* London: C40 Cities Leadership Group. Available at: www.resourcecentre.c40.org (Accessed 07/11/2020).

C40 Cities (2018c) *Inclusive Planning.* London: C40 Cities Leadership Group. Available at: www.resourcecentre.c40.org (Accessed 07/11/2020).

C40 Cities (2018d) *C40 CITIES CLIMATE CHANGE RISK ASSSESMENT GUIDANCE.* London: C40 Cities Leadership Group. Available at: www.resourcecentre.c40.org (Accessed 07/11/2020).

Dantzig, G. & Saaty, T. (1973) *Compact City: Plan for a Liveable Urban Environment*. San Francisco: W. H. Freeman.

Dawson, R. (2007) 'Re-engineering cities: a framework for adaptation to global change', *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 2007 365 (1861), pp. 3085.

Dawson, R.J. (2011) 'Potential pitfalls on the transition to more sustainable cities and how they might be avoided', *Carbon Management*, 2011 2(2), pp. 175-188.

Defra (2005) *The UK Government Sustainable Development Strategy*. UK. Available at: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/69412/pb10589-securing-the-future-050307.pdf (Accessed: 07/11/2020).

DCLG (2011) *National Planning Policy Framework*. UK. Available at: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/11802/1951747.pdf (Accessed 07/11/2020).

Echenique, M , H., Flowerdew, A, D, J., Hunt, J,D., Mayo, T, R., Skidmore, I, J., and Simmonds, D, C. (1990) 'The MEPLAN models of Bilbao, Leeds and Dortmund', *Transport Reviews*, 1990 10:4, pp. 309-322.

Epstein, J, M. (2008) 'Why Model?', *Journal of Artificial Societies and Social Simulation,* 2008 11(4).

Ford, A, Barr, S, Dawson, R, and James, P. (2015) 'Transport Accessibility Analysis Using GIS: Assessing Sustainable Transport in London.' *ISPRS International Journal of Geo-information* 2015 4.1, pp. 124-49.

Ford, A., Dawson, R., Blythe, P. and Barr, S. (2018) 'Land-use transport models for climate change mitigation and adaptation planning', *Journal of Transport and Land Use*, 2018 11(1), pp. 83-101.

Forrester, J.W. (1969) *Urban dynamics*. Cambridge, MA.: M.I.T. Press.

Fünfgeld, H. (2010) 'Institutional Challenges to Climate Risk Management in Cities.' *Current Opinion in Environmental Sustainability,* 2010 2, pp. 156-60.

Gasparatos, A., El-Haram, M. & Horner, M., (2008). 'A critical review of reductionist approaches for assessing the progress towards sustainability'. *Environmental Impact Assessment Review*, 2008 28, pp. 286–311.

Geertman, S. & Stillwell, J. (2009). *Planning Support Systems Best Practice and New Methods*. Dordrecht: Springer.

Gibson, R.B. (2006) 'Beyond the Pillars: Sustainability Assessment as a Framework for Effective Integration of Social, Economic and Ecological Considerations in Significant Decision-Making'. *Journal of Environmental Assessment Policy and Management*, 2006 08(03), pp.259–280.

GLA (2008) *The London Plan*.UK. Available at: https://www.london.gov.uk/what-we-do/planning/london-plan/past-versions-and-alterations-london-plan/london-plan-2008 (Accessed: 07/11/2020).

GLA (2016) *The London Plan*. UK. Available at: https://www.london.gov.uk/what-we-do/planning/london-plan/current-london-plan/london-plan-2016-pdf (Accessed: 07/11/2020).

GLA (2018): *'Zero carbon London: A 1.5C compatible plan'*. UK. Available at: https://www.london.gov.uk/sites/default/files/1.5c_compatible_plan.pdf (Accessed: 07/11/2020).

Granell, C, Schade, S, and Ostländer, N (2013) 'Seeing the Forest through the Trees: A Review of Integrated Environmental Modelling Tools.' *Computers, Environment and Urban Systems*, 2013 41, pp. 136-50.

Hall, J.W., Dawson, R.J., Walsh, C.L., Barker, T., Barr, S.L., Batty, M., Bristow, A.l., Burton, A., Carney, S., Dagoumas, A., Evans, S., Ford, A.C., Glenis, V., Goodess, C.G., Harpham, C, Harwatt, H., Kilsby, C., Köhler, J., Jones, P., Manning, L., McCarthy, M., Sanderson, M., Tight, M. and Zanni, A.M. (2009) *Engineering Cities. How can cities grow whilst reducing emissions and vulnerability?* Newcastle UK: Newcastle University.

Hamilton, S, H, Sondoss, E, Guillaume, J, H, A, Jakeman, A, J, and Pierce, S, A. (2015) 'Integrated Assessment and Modelling: Overview and Synthesis Of salient Dimensions.' *Environmental Modelling and Software*. 2015. pp. 215-29.

Hansen, W.G. (1959) 'How Accessibility Shapes Land Use', *Journal of the American Institute of Planners*, 1959 25(2), pp. 73-76.

Hägerstraand, T. (1970) 'WHAT ABOUT PEOPLE IN REGIONAL SCIENCE?', *Papers in Regional Science*, 1970 24 1, pp. 7-24.

Hargreaves, AJ. (2015) 'Representing the dwelling stock as 3D generic tiles estimated from average residential density.', *Computers, Environment and Urban Systems* 2015 54. pp. 280–300.

Harlan, S, L, and Ruddell, D, M. (2011) 'Climate Change and Health in Cities: Impacts of Heat and Air Pollution and Potential Co-benefits from Mitigation and Adaptation.' *Current Opinion in Environmental Sustainability*, 2011 3. pp. 126-34.

Heidrich, O., Reckien, Olazabal, Foley, Salvia, De Gregorio Hurtado, Orru, Flacke, Geneletti, Pietrapertosa, Hamann, Tiwary, Feliu, and Dawson. (2016) ‘National Climate Policies across Europe and Their Impacts on Cities Strategies.’, *Journal of Environmental Management.* 2016 168. pp. 36-45.

Hodson, M., Marvin, S., Robinson, B. and Swilling, M. (2012) 'Reshaping Urban Infrastructure', *Journal of Industrial Ecology*, 2012 16 (6), pp. 789-800.

Howard, E., (1902). *Garden Cities of Tomorrow*. London, UK: Swan Sonnenschein & Company.

Hunt, A. and Watkiss, P. (2011) 'Climate change impacts and adaptation in cities: a review of the literature', *Climatic Change*, 2011 104(1), pp. 13-49.

IPCC, (2014). *Climate Change 2014: Impacts, Adaptation, and Vulnerability. Part A: Global and Sectoral Aspects. Contribution of Working Group II to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge: Cambridge University Press. Available at: https://www.ipcc.ch/report/ar5/wg2/ (Accessed: 07/11/2020).

Jenkins, K., Hall, J., Glenis, V., Kilsby, C., McCarthy, M., Goodess, C., Smith, D., Malleson, N. and Birkin, M. (2014) 'Probabilistic spatial risk assessment of heat impacts and adaptations for London', *Climatic Change*, 2014 124(1-2), pp. 105-117.

Knapen, R, Janssen, S, Roosenschoon, O, Verweij, P, Winter, W, Uiterwijk, M, and Wien, J-E. (2013) ‘Evaluating OpenMI as a Model Integration Platform across Disciplines.’, *Environmental Modelling and Software* 2013 39. pp. 274-82.

Larson, W., Liu, F. and Yezer, A. (2012) 'Energy footprint of the city: Effects of urban land use and transportation policies', *Journal of Urban Economics*, 2012 72, pp. 147-159.

LEGGI (2020) *London Energy and Greenhouse Gas Inventory*. Available at: https://data.london.gov.uk/dataset/leggi (Accessed: August 23, 2020).

Lowry, I.S. (1964) *A model of metropolis*. Santa Monica, CA: Rand Corporation, Santa Monica, CA.

McDonnell, S. and Zellner, M. (2011) 'Exploring the effectiveness of bus rapid transit a prototype agent-based model of commuting behaviour.', *Transport Policy*, 2011 18(6), pp. 825-835.

McEvoy, D., Ahmed, I. and Mullett, J. (2012) 'The impact of the 2009 heat wave on Melbourne's critical infrastructure', *Local Environment*, 2012 17(8), pp. 783-796.

McFadden, D., (1973). *Conditional logit analysis of qualitative choice behaviour*. In *Analysis of Qualitative Choice Behaviour*. Berkeley, California: Institute of Urban and Regional Development.

McKinsey Global Institute (2010) *India's urban awakening: Building inclusive cities, sustaining economic growth*. Available at: https://www.mckinsey.com/~/media/mckinsey/featured%20insights/urbanization/urban%20awakening%20in%20india/mgi_indias_urban_awakening_full_report.ashx (Accessed 07/11/2020).

McKinsey Global Institute (2011) *Urban World: Mapping the economic power of cities*. Available at: https://www.mckinsey.com/~/media/McKinsey/Featured%20Insights/Urbanization/Urban%20world/MGI_urban_world_mapping_economic_power_of_cities_exec_summary.pdf (Accessed: 07/11/2020).

Melia, S., Parkhurst, G. & Barton, H., (2011). 'The Paradox of Intensification', *Journal of Transport Policy*, 2011 18(1), pp.46–52.

Mikovits, C., Rauch, W. and Kleidorfer, M. (2014) 'Dynamics in Urban Development, Population Growth and their Influences on Urban Water Infrastructure', *Procedia Engineering*, 2014 70, pp. 1147-1156.

Mikovits, C., Rauch, W. and Kleidorfer, M. (2015) 'A dynamic urban development model designed for purposes in the field of urban water management', *Journal of Hydroinformatics*, 2015 17(3), pp. 390-403.

Mikovits, C., Rauch, W. and Kleidorfer, M. (2017) 'Importance of scenario analysis in urban development for urban water infrastructure planning and management', *Computers, Environment and Urban Systems*. 2018 68. pp. 9-16.

Moekel, R, Spiekermann, K, Schürmann, C and Wegener, M. (2013) 'Microsimulation of Land Use', *International Journal of Urban Sciences*. 2013 7. pp. 14-31.

Nicholls, R.J., Wong, P.P., Burkett, V., Codignotto, J., Hay, J., McLean, R., Ragoonaden, S., Woodroffe, C.D., Abuodha, P.A.O. and Arblaster, J. (2007) 'Coastal systems and low-lying areas.' in Parry, ML, Canziani, OF, Palutikof, JP, van der Linden, PJ, and Hanson, CE *Climate change 2007: impacts, adaptation and vulnerability*. Cambridge, UK: Cambridge University Press. Available at: https://ro.uow.edu.au/cgi/viewcontent.cgi?article=1192&context=scipapers (Accessed: 07/11/2020).

ODPM - Office of the Deputy Prime Minister (2004) *Sustainability Appraisal of Regional Spatial Strategies and Local Development Frameworks Consultation Paper.* London: Available at: http://www.unece.org/fileadmin/DAM/env/eia/documents/SEAguides/UK%20Sustainability%20Appraisal%20for%20Regional%20Strategies%20consultation%20paper.pdf (Accessed: 07/11/2020).

Pearce, D., Markandya, A. and Barbier, E.B. (1989) *Blueprint for a green economy*. London: Earthscan.

Pregnolato, M., Ford, A., Robson, C., Glenis, V., Barr, S., and Dawson, R. (2016) 'Assessing Urban Strategies for Reducing the Impacts of Extreme Weather on Infrastructure Networks.', *Royal Society Open Science,* 3: 160023. doi: 10.1098/rsos.160023.

Randolph, J. (2013) 'A Review of "Green Cities of Europe: Global Lessons on Green Urbanism.', *Journal of the American Planning Association*, 79(1), pp.101–102.

Reckien, D., Salvia, M., Heidrich, O., Church, J, M., Pietrapertosa, F., De Gregorio-Hurtado, S., D'Alonzo, V., Foley, A., Simoes, S, G., Krkoška Lorencová, E., Orru, H., Orru, K., Wejs, A., Flacke, J., Olazabal, M., and Geneletti, D. (2018) 'How Are Cities Planning to Respond to Climate Change? Assessment of Local Climate Plans from 885 Cities in the EU-28.', *Journal of Cleaner Production* 191.C (2018): pp. 207-19.

Rockström, J., Gaffney, O., Rogelj, J., Meinshausen, M., Nakicenovic, N., and Schellnhuber, H, J. (2017) 'A Roadmap for Rapid Decarbonization.', *Science,* 355.6331 (2017), pp. 1269-1271.

Salvini, P., and Miller, E. (2005) 'ILUTE: An Operational Prototype of a Comprehensive Microsimulation Model of Urban Systems.', *Networks & Spatial Economics,* 5.2 (2005): pp. 217-234.

Santé, I., García, A.M., Miranda, D. and Crecente, R. (2010) 'Cellular automata models for the simulation of real-world urban processes: A review and analysis', *Landscape and Urban Planning*, 96(2): pp. 108-122.

Schelling, T.C. (1971) 'Dynamic models of segregation', *Journal of mathematical sociology*, 1(2), pp. 143-186.

Silva, E.A, and Clarke, K.C. (2002) 'Calibration of the SLEUTH Urban Growth Model for Lisbon and Porto, Portugal.', *Computers, Environment and Urban Systems*, 26.6 (2002): pp. 525-552.

Simmonds, D C. (1999) 'The Design of the Delta Land-Use Modelling Package.', *Environment and Planning B: Planning and Design*, 26.5 (1999): pp. 665-684.

Spiekermann, K. and Wegener, M., (1999). ‘Freedom from the tyranny of zones: Towards new GIS-based spatial models.’ in Fotheringham, A, S., and Wegener, M. *Spatial Models and GIS: New Potential and New Models*. London: Taylor and Francis, pp.45-61.

Small, C. and Nicholls, R. (2003) 'A Global Analysis of Human Settlement in Coastal Zones', *Journal of Coastal Research*, 19(3), pp. 584-599.

TFL (2006) *London Travel Report 2006*. London: Transport for London, Available at: http://content.tfl.gov.uk/London-Travel-Report-2006-final.pdf (Accessed: 07/11/2020).

TFL (2007) *London Travel Report 2007*. London: Transport for London, Available at: http://content.tfl.gov.uk/London-Travel-Report-2007-final.pdf (Accessed: 07/11/2020).

TFL (2010) *Measuring Public Transport Accessibility Levels*. London: Transport for London, Available at: https://s3-eu-west-1.amazonaws.com/londondatastore-upload/PTAL-methodology.pdf (Accessed: 07/11/2020).

TFL (2013) *London Congestion Charge information*, London: Transport for London. Available at: http://www.tfl.gov.uk/roadusers/congestioncharging/6709.aspx (Accessed: December 2013).

UCLG (2015) *Paris City Hall Declaration*. Available at: https://www.uclg.org/sites/default/files/climate_summit_final_declaration.pdf (Accessed: 07/11/2020).

UN (2012) *Urban Planning for City Leaders*. Available at: https://unhabitat.org/sites/default/files/download-manager-files/UN%20Habitat%20UPCL%2014-02624%20-%20Combine.pdf (Accessed: 07/11/2020).

UNFCC (2015) *Adoption of the Paris Agreement*. Available at: https://unfccc.int/resource/docs/2015/cop21/eng/l09r01.pdf (Accessed: 07/11/2020).

Voinov, A., and Bousquet, F. (2010) ‘Modelling with Stakeholders.’, *Environmental Modelling and Software*, 25.11 (2010), pp. 1268-1281.

Voinov, A. and Shugart, H.H. (2013) '‘Integronsters’, integral and integrated modeling', *Environmental Modelling and Software*, 39, pp. 149-158.

Waddell, P., Borning, A., Noth, M., Freier, N., Becke, M. and Ulfarsson, G. (2003) 'Microsimulation of Urban Development and Location Choices: Design and Implementation of UrbanSim', *Networks and Spatial Economics*, 3(1), pp. 43-67.

Waddell, P. (2011) 'Integrated Land Use and Transportation Planning and Modelling: Addressing Challenges in Research and Practice', *Transport Reviews*, 31(2), pp. 209-229.

Waddell, P., Garcia-Dorado, I., Maurer, S.M., Boeing, G., Gardner, M., Porter, E. and Aliaga, D. (2018) 'Architecture for Modular Microsimulation of Real Estate Markets and Transportation', *Symposium on Applied Urban Modelling*, Cambridge UK, 27/06/2018. Available at: https://arxiv.org/pdf/1807.01148.pdf (Accessed: 07/11/2020).

Walker, W. T., Gao, S., and Johnston, R, A. (2007) 'UPlan: Geographic Information System as Framework for Integrated Land Use Planning Model.', *Transportation Research Record,* No. 1994, pp. 117-127.

Walsh, C, L., Dawson, R, J., Hall, J, W., Barr, S, L., Batty, M., Bristow, A, L., Carney, S., Dagoumas, A, S., Ford, A, C., Harpham, C., Tight, M, R., Watters, H., and Zanni, A, M. (2011) 'Assessment of Climate Change Mitigation and Adaptation in Cities.', *Urban Design and Planning*, 164.2 (2011), pp. 75-84.

Walsh, C, L., Roberts, D., Dawson, R, J., Hall, J W., Nickson, A., and Hounsome, R. (2013) 'Experiences of Integrated Assessment of Climate Impacts, Adaptation and Mitigation Modelling in London and Durban.', *Environment & Urbanization*, 25.2 (2013), pp. 361-380.

WebTAG (2009) *Transport Analysis Guidance*, Available at: http://www.dft.gov.uk/webtag/ (Accessed: 01/09/2009).

Wegener, M. (2011) 'From Macro to Micro—How Much Micro is too Much?', *Transport Reviews*, 31(2), pp. 161-177.

Wegener, M. (2011a). *The IRPUD model*. Dortmund, Germany: Spiekermann & Wegener Urban and Regional Research. Available at: http://spiekermann-wegener.de/mod/pdf/AP_1101_IRPUD_Model.pdf (Accessed: 07/11/2020).

Wegener, M. (2014) 'Land-use transport interaction models.' in Fischer, M, and Nijkamp, P. *Handbook of Regional Science*. Berlin: Springer-Verlag, pp. 741–758.

Williams, K., (2004) 'Can Urban Intensification Contribute to Sustainable Cities? An International Perspective.', *City Matters, Official Electronic Journal of Urbanicity, UN Habitat Partnership Initiative*. Available at: http://unpan1.un.org/intradoc/groups/public/documents/APCITY/UNPAN026009.pdf (Accessed 06/01/2012).

Wilson, A.G. (1970) *Entropy in urban and regional modelling*. London: London, Pion.