Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

CITYSCAPE: A CITY-LEVEL DIGITAL TWIN MODEL GENERATOR FOR SIMULATION & ANALYSES

Dhananjai M. Rao

CSE Department
Miami University
510 E. High Street
Oxford, OH 45056, USA

ABSTRACT

Cities and large urban areas face a myriad of challenges ranging from city planning, developing sustainable transportation, managing natural catastrophes, and mitigating communicable diseases. Addressing these challenges requires effective analysis and planning which in turn necessitates the use of sufficiently detailed models or "digital twins." Such detailed models that embody multifaceted demographic and city characteristics are challenging to generate. This paper presents our ongoing work to develop a novel model generation method and software suite called CITYSCAPE, that fuses diverse real-world data sets to generate a digital twin for a given city. Specifically, our method combines data from authoritative sources including PUMS, PUMAs, and OpenStreet Map to generate the digital twin. We have used the city of Chicago (IL, USA) as a case study to verify and validate (with ~85% confidence) our proposed method.

1 INTRODUCTION

Increasing urbanization and population growth pose a broad range of challenges ranging from city planning, developing sustainable transportation, managing natural catastrophes, and mitigating communicable diseases. For example, during the COVID-19 pandemic, several cities around the world faced challenges in designing effective policies to contain the epidemic through strict quarantines while providing necessary food and resources to people (Rao and Rao 2021; Wang et al. 2021). Another example of poor planning was witnessed in the Gulf Coast of the United States during the 2005 hurricane Katrina – it was tragically discovered the evacuation plans were not adequate to evacuate the impacted area (Daniels 2007). As discussed by Daniels (2007), the root issue is that responses to particular catastrophes are usually based on historical examples of similar events and not on current scenarios. Conventional statistical analyses are not conducive for "what-if" type analyses that are required for policy assessments. Furthermore, they do not yield sufficiently detailed and intuitive information about households and people living in a city. Hence, it is imperative that effective methods and tools are developed so that we are prepared to face the growing challenges posed by various factors, including climate change and emergent communicable diseases. Otherwise, as Benjamin Franklin said – "By failing to prepare, you are preparing to fail." Hence, innovative approaches are needed to proactively address these growing challenges.

Effective analysis and planning require sufficiently detailed models. Modeling and simulation-based methods are widely used for the study and analysis of complex dynamics of human populations. Since, simulations rely on valid, comprehensive, and robust models, modeling plays a pivotal role (Schmidt and Rao 2017; Giridharan and Rao 2016). Such models are also called "digital twins" as they provide a more realistic characterization of the real world (Jones et al. 2020). As discussed by Schmidt et al. (Schmidt and Rao 2017) et al., digital twins of population demographics need to embody several key characteristics, namely, (a) *Realism*: The model must be realistic and mirror geographic, demographic, and behavioral characteristics; (b) *Reusability & accuracy*: Investments into model development and validation

979-8-3503-6966-3/23/$31.00 ©2023 IEEE

are effectively amortized only when models can be reused or easily adapted for different types of analyses; and (c) *Computational costs*: Time and resources required for model generation, validation, simulation, and analysis need to be balanced with realism, accuracy, and effective use (Chen and Zhan 2008). However, generating comprehensive models for human populations continues to remain a challenging task (Barrett et al. 2009; Schmidt and Rao 2017). Very few detailed models are readily available for use by various modeling and simulation communities (Schmidt and Rao 2017).

1.1 Overview and Key Contributions of This Work

This paper presents our ongoing work to develop a novel model generation method that fuses diverse real-world data sets to generate a digital twin for a given city. Specifically, our method combines data from authoritative sources including Public Use Microdata Sample (PUMS), Public Use Microdata Areas (PUMAs), and detailed street maps to generate the digital twin for a given city. Our method has been incorporated into a software suite called CITYSCAPE. The software system also includes tools for visualizing the input data and the generated model. Section 3 presents the details of our methodology and CITYSCAPE. Our methodology and software suite are the key contributions of this work. Section 2 compares and contrasts our methods with other closely related investigations. In this study, we have used the city of Chicago (IL, USA) as a case study to verify and validate our method and software suite. Relevant aspects of our case study are discussed in Section 4. Section 5 provides concluding remarks along with our plans for future work.

2 RELATED WORKS

The focus of this paper is our method for the generation of a digital twin model of the human population in a given city. The generation of realistic and comprehensive models of human populations is an active research area in diverse fields, such as computational epidemiology, sustainable transportation, and socioeconomics. In such models, humans are either modeled as a collection of interacting individuals or groups. Group models represent a collection of collocated individuals modeled as an indivisible unit. Different types of group models have been proposed by several investigators including (Balcan et al. 2010; Keeling 2005; Rao et al. 2009). In their models, the groups are organized based on their geographic locations resulting in a logical structure akin to a Voronoi tessellation. Interactions between groups are modeled implicitly based on their adjacency or via explicit mobility networks. The benefit of using a group model is it reduces the computational cost because the model consists of fewer entities which significantly reduces computations performed at each time step. Furthermore, such models do not require detailed, voluminous data about the population, which can be hard to obtain. The primary disadvantage of aggregate or group models is that information about each individual is not preserved. The models do not preserve heterogeneity that may be present within the group. However, such information may be vital for certain types of analyses and the design of public policies.

Consequently, several researchers have proposed the use of individual-based models, where each individual is independently modeled. Such models essentially embody contact networks that define temporospatial interactions that occur between individuals. Longini et al. (2005) discuss the generation and use of synthetic, individual-based models for containing influenza epidemics. They use a variety of data sources to generate individuals and their temporospatial activities. Bhatele et al. (2017) discuss enhancements to modeling and simulation of the individual-based model generated as part of prior work by Barrett et al. (2009). Several investigations of the COVID-19 pandemic have used individual-based models for country-level analyses of the epidemic and the effectiveness of various interventions (Unwin et al. 2020; Verity et al. 2020). The advantages of individual-based models are that they are less prescriptive and provide a more realistic digital-twin model. However, the shortcomings are that they are harder to generate, calibrate, verify, and validate. Moreover, individual-based models can be computationally demanding, often requiring supercomputers to enable simulations within acceptable run times (Barrett et al. 2009).

2.1 Similarities and Differences of Our Method from Prior Works

Prior investigations typically focus on either group-based or individual-based models. In contrast, our digital-twin model generation method aims to preserve the advantages of both the group-based and individual-based models with scope for further aggregation. Similar to the aforementioned group-based models, in our proposed method (see Section 3.4), we generate "households" (or families) that represent a group of people living together. In addition, we also include individual information with each “household” that is generated. In other words, our method yields models that can be used either at a group level or at an individual level based on the type of static or simulation-based analysis to be conducted. Moreover, our method also preserves geographical annotations that can be used to further aggregate the model to analyze or simulate larger groups. Similar to models by Barrett et al. (2009), modelers can also include any of the >100 different household and people attributes from PUMA data into the model to facilitate different analyses. However, unlike their models, our generated model also includes detailed roadways and building information that is required for certain types of geospatial analyses. In contrast to some of the prior research, our software suite called CITYSCAPE is multithreaded to enable rapid model generation and visualization of our models. Our model generator produces visualization using `XFig` text format, which eases editing and generating visualizations. This feature of generating visualization as part of model generation distinguishes CITYSCAPE from several of the prior works discussed earlier. The aforementioned combination of features collectively distinguish our work form prior closely-related investigations. The software and generated models are publicly available for use by the scientific community at github.com/raodj/cityscape.

3 METHODOLOGY

An overview of the model generation methodology used by CITYSCAPE is shown in Figure 1. As illustrated by the figure, our proposed method fuses diverse data from different sources to generate a comprehensive digital-twin model for a given city. The model generation process proceeds in two key phases that are discussed in detail in the following subsections.

3.1 Phase 1: Roadways & Building Data from Open Street Map (OSM)

The first step of the process begins with the modeler downloading the freely available, community-maintained, Open Street Map (OSM) data (OpenStreetMap contributors 2023) for a given city. Alternatively, data from other sources such as Google Maps can be used, but the data is not freely available and hence we propose to use OSM datasets. Typically, the data is downloaded using a bounding rectangle around the city of interest. However, the city boundaries are seldom rectangular. Hence, the downloaded OSM data at this phase include additional areas that may not be relevant to the city of interest. Consequently, in our method,

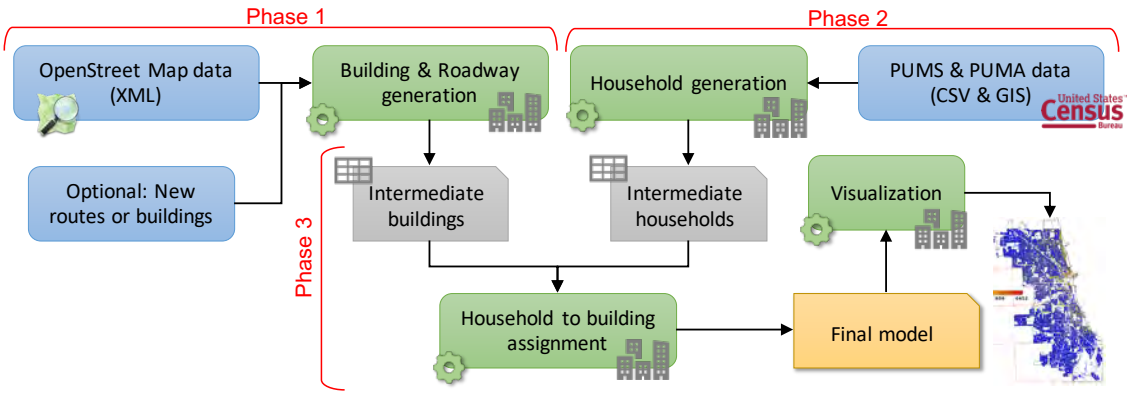


Figure 1: Overview of model generation methodology used in CITYSCAPE.

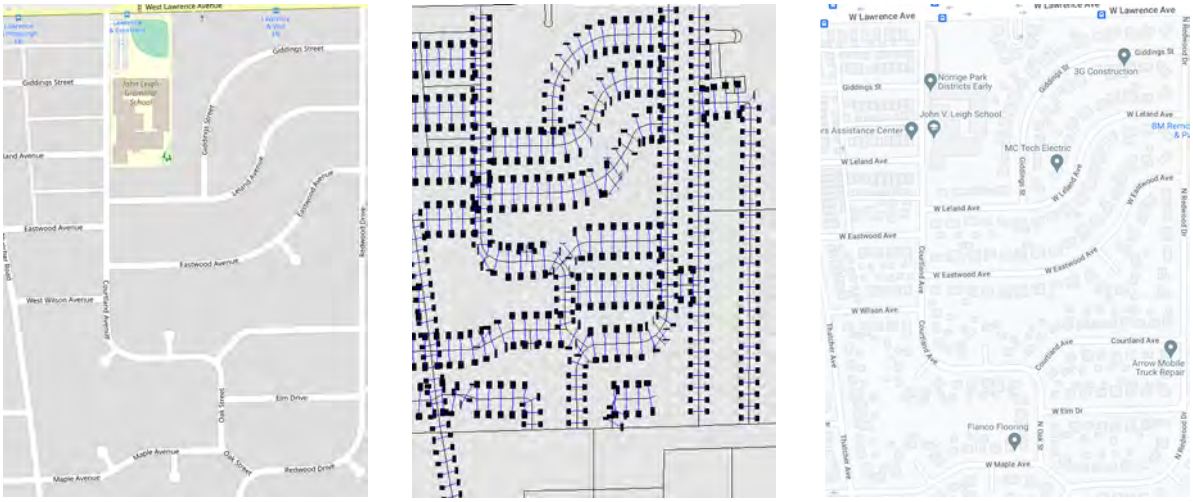
other data sources used in subsequent model generation phases can be used to further fine-tune the city boundaries. This approach is a compromise between a modeler's time to define the geographic boundaries manually (*i.e.,*, rectangular regions are easier to specify than complex polygons for city bounds) versus a slight increase in computational cost for processing additional data that may not be used for generating the final model.

3.1.1 Phase 1.1: Generation of Existing & New Roadways from OSM Data

The OSM data is an XML file that includes detailed information about roadways in the region along with metadata about the roadways, such as their `kind` (e.g., highways, secondary ways, etc.), their speed limits, etc. A roadway (or a "way") is represented as a set of interconnected nodes. Nodes are represented as geographic points with specific latitudes and longitudes. Intersecting ways have a common node for the intersections. At this step, a modeler also has the option to add new nodes and roadways to the generated model. This feature enables modelers to analyze the impacts of new roadways or proposed extensions in the city. Currently, the OSM data for certain roadways is incomplete with missing metadata for the speed limit of roadways. In such scenarios, the speed limit is inferred based on its `kind`.

3.1.2 Phase 1.2: Generation and Verification of Existing & Synthetic Buildings

In addition to roadway data, the OSM data also provides metadata information about buildings along with their shapes represented as a polygon. Each vertex in this polygon is denoted by its latitude and longitude. The metadata for a building includes its `type` (i.e., `house`, `church`, `school`, `office`, etc.), the number of levels in the building, and several `tags` that provide additional information. The metadata and `tags` are used to classify buildings are residential or non-residential buildings. In addition, the shapes of buildings and the number of levels are used to compute the area of buildings. This information is used to assign households to residential buildings in subsequent steps. In certain areas, the OSM data for buildings, particularly homes, is missing. Figure 2(a) illustrates an example of such an area in the city of Chicago, IL, USA. Consequently, in these areas, synthetic buildings representing homes are automatically generated by CITYSCAPE along residential roadways that do not have any buildings, as shown in Figure 2(b). The generation of synthetic buildings is also applied to any new roadways the modeler may have added in the



(a) OpenStreet Map (b) Generated homes (c) Google Maps

Figure 2: Region in the city of Chicago (IL, USA) where synthetic homes were generated to compensate for missing building data along with Google Map data used for V&V.

previous step. The size of the homes and the spacing between them is specified by the modeler to reflect typical homes in the area. The synthetic homes are generated on either side of a roadway as shown in Figure 2(b). In our final model, all buildings are simplified to rectangular regions. In addition, in this step, entryways for each building are computed based on the nearest roadway to the building.

Verification and Validation (V&V): We have manually verified and validated the proposed building generation process using the city of Chicago as a case study. For verification, we have used Google Maps (see Figure 2(c) for example), which has more complete data when compared to OpenStreet Map. For the verification process, CITYSCAPE's map generation feature was used to generate map fragments as shown in Figure 2(b). The map fragments were then visually compared with the corresponding region in Google Maps. Our synthetic building generation process yields good results except in certain cases where other types of buildings such as offices are present in certain areas. However, a modeler can manually add such buildings to the model as well. Additional V&V of our proposed methods are discussed in Section 4.

3.1.3 Phase 1.3: Specification of Accurate City Boundaries

As discussed earlier in this subsection, the OSM data for a city is approximately downloaded using a rectangular bounding box around the city limits. This reduces the manual overheads for a modeler. The approximate rectangular region can be fine-tuned by a modeler by providing accurate city boundaries in the form of Geographic Information Systems (GIS) shape files. An example of such a shape file for the city of Chicago, IL, USA is shown in Figure 3. It must be noted that the figure is generated by CITYSCAPE using the supplied GIS shape files. The GIS shape files are available for public use from the city, state, and country-level government organizations (The City of Chicago 2023b). The shape file data is intersected with the OSM data and only roadways and buildings within the bounds of all the shapes are included in the final model generated by CITYSCAPE. In addition, the boundary shapes are also used to determine population distributions from other data sources as discussed in Section 3.2

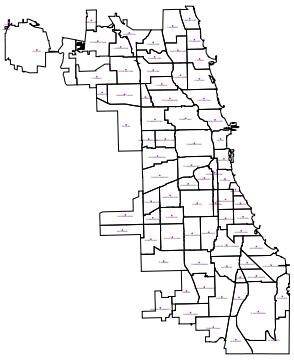


Figure 3: Example GIS boundaries.

3.2 Phase 2: Household and Population Generation

The second phase of the digital-twin model generation focuses on creating and assigning households to the residential buildings in the model. Currently, CITYSCAPE provides two different approaches for generating populations, namely (a) an approximate household generation approach based on gridded population data, and (b) a more precise household generation approach using census micro-samples. These two approaches are further discussed in the following two subsections.

3.3 Household Generation Using Gridded Population Data

Gridded population data are widely available and used for modeling. Common sources of gridded population data include data sets from NASA's Socioeconomic Data and Applications Center (SEDAC) (CIESIN 2016) and LandScan™ data set from the Oak Ridge National Laboratory (ORNL) (Oak Ridge National Laboratory (ORNL) 2012). Gridded population data are typically made available as raster Geographic Information Systems (GIS) files. Each grid is of fixed size and provides information about the estimate of the population in the given grid. The grids of interest are determined by intersecting them with a given city's shape boundaries as discussed in Section 3.1.3. An example, of gridded population data for the city of Chicago, is shown in Figure 4. The population in each grid is used to

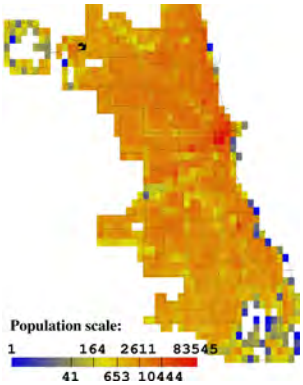


Figure 4: Example gridded population data.

divide the population based on the number of buildings and the size of buildings. This approach does not require any additional data. However, the compromise is that the generated population distribution is very approximate. Such a model may be sufficient for certain types of analyses but insufficient for detailed analyses based on households and the distribution of households. Consequently, CITYSCAPE also enables population generation using a more involved process as discussed in Subsection 3.4.

3.4 Household Generation Using Public Use Microdata Sample (PUMS)

CITYSCAPE also provides a comprehensive method for generating and assigning households to buildings using the Public Use Microdata Sample (PUMS) available from the US Census Bureau (United States Census Bureau 2023b). PUMS data is widely used for different types of socioeconomic analyses. The PUMS data provides a statistical summary of the population based on census responses to the American Community Survey (ACS).

3.4.1 Sources of Errors in the PUMS Data When Compared to Full Census Data

Although PUMS aims to provide an accurate summary of the demographics, there are a few sources of errors to be aware of, namely:

1. It must be noted that the PUMS data is a summary and certain appropriate changes are also made to anonymize the data and preserve confidentiality (United States Census Bureau 2023b).
2. The PUMS microdata is a sample of the full ACS microdata and includes only about two-thirds of the records that are used to produce ACS estimates.
3. PUMS data is available either as a 1-year or 5-year estimate. The 1-year estimate is more recent but is based on ~1% of the population. On the other hand, the 5-year data is based on ~5% of the population but lags behind in current estimates. These two data sets provide a tradeoff between recentness and accuracy.

Consequently, the summary statistics (such as average household size) characteristics of the model generated using PUMS data are expected to slightly deviate from the corresponding summary statistics in the full US Census data. In the case of Chicago, the number of households in the generated model is fewer by ~4% (*i.e.,* 1.06 million instead of 1.11 million) and the average household size is lower by ~6% (*i.e.,* 2.27 persons per household instead of 2.41). A modeler has the option to suitably scale the PUMS data to obtain a closer match to the summary statistics from the complete census data.

3.4.2 PUMS Data Set Details

The PUMS data for a city are distributed as Comma Separated Values (CSV) text files and include two types of files: (a) one CSV file for "person" records, and (b) one for "housing" unit records. Each record in the person file represents a single person with over 100 attributes for each person, such as age, income, weight, health parameters, etc. Each person is also associated with a housing unit record to enable the generation of households. Currently, CITYSCAPE only uses household records in PUMS. It does not use group quarters (GQ) records that represent people in prisons, nursing homes, or college dormitories. Since GQ records are not used, this also causes some of the differences discussed in the previous paragraph. We plan to include GQ records as well in the near future.

3.4.3 Combining PUMS data with Public Use Microdata Areas

The PUMS data provides a summary of people and households within a given geographic region called Public Use Microdata Area (PUMA) (United States Census Bureau 2023a). PUMAs are non-overlapping, statistical geographic areas that partition each state or equivalent entity into geographic areas containing no fewer than 100,000 people. The US Census Bureau defines PUMAs for the tabulation and dissemination of decennial

census and PUMS data. Each PUMS record is associated with a PUMA region. The PUMA regions are provided in the form of GIS vector shape files. However, the PUMA regions do not follow the same geographic boundaries as city limits as illustrated in Figure 5. Hence, additional processing is necessary to detect overlap between city boundaries (see Section 3.1.3) and PUMA regions. CITYSCAPE includes custom computational geometry algorithms to detect intersections between polygons and compute percentage overlaps. The percentage overlap is used to appropriately scale the number of PUMS households generated in overlapping areas.

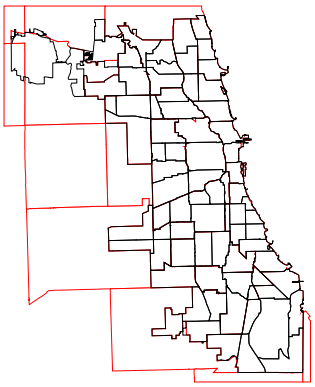


Figure 5: PUMA regions (in red) for Chicago.

3.5 Phase 3: Assigning Households to Buildings

The final phase of model generation assigns households generated in Phase 2 to residential buildings generated in Phase 1 (see Section 3.1) for each PUMA region. First, the households are sorted based on the type of building they are associated with in the PUMS record. The buildings are sorted based on their square footage and the assignment process uses these two sorted lists. For example, single-family households associated with independent houses are assigned first. Next, households that live in apartments and condominiums are assigned. From our experiments, we find this approach provides a sufficiently realistic mapping between households and the type of building they live in. As discussed in Section 4, we have manually verified several randomly selected households to verify and validate the final model generated at the end of this phase.

3.6 Technical Details on Implementation of CITYSCAPE

Our model generation process involves three distinct phases that interact with each other. Each phase operates on different types of data, including CSV files, GIS files, and other custom text files for fine-tuning the model. CITYSCAPE uses `gdal` library for reading GIS shape files. The shape files are then converted to polygons or rings for further processing using custom computational geometry algorithms. Custom readers are used for loading CSV and other text files. The data processed in each phase are stored in memory using custom data structures to streamline model generation. Consequently, model generation uses a lot of memory – for example, generating the model of Chicago requires ~6.6 GB of RAM. Hence, careful control over memory usage is critical. Moreover, the computational geometry algorithms take time to run on large data sets with many complex polygons. Accordingly, to have better control over memory usage and ensure good performance, we have implemented CITYSCAPE in C++17. We have used the object-oriented features of the language to encapsulate data and provide convenient and quick access to related information. The outputs from the model generation phase is stored in a simple text file format. The objective is to ease reading and processing of models using different tools and programming languages. CITYSCAPE also includes tools to generate different figures in XFig file format that can then be converted to other formats such as PNG, JPEG, or PDF. The figures in this paper have been produced by CITYSCAPE.

3.6.1 Parallelizing Model Generation

To further accelerate model generation, we have also multithreaded computationally demanding parts of the process using OpenMP. Specifically, we have multithreaded the following parts of model generation:

- processing OSM XML to extract nodes, ways, and buildings,
- creation of synthetic homes (see Section 3.1.2),
- loading and detection of intersecting GIS polygons which is part of the computational geometry routines in CITYSCAPE, and
- creation and distribution of households using PUMS and PUMA data.

Our multithreaded approach substantially reduces model generation and visualization times.

4 CASE STUDY: CITY OF CHICAGO

In this study, we have used the City of Chicago (IL, USA) as a case study and to verify and validate the model generation methodology of CITYSCAPE. Chicago has been chosen as the city of interest for several reasons:

1. It is the third largest city in the USA with a population of ~2.69 million and serves as a good test case for CITYSCAPE.
2. The city boundary shapes, PUMS, and PUMA data are readily available.
3. The City of Chicago provides several data sets that are used for verification and validation of the generated model (The City of Chicago 2023a).
4. Availability of additional data sets (such as accident data, taxi cab ride data, etc.) provide an opportunity for developing novel applications using the generated model.

The different data to be used for model generation were obtained from the following data sources: (a) Open Street Map XML data (910 MB) was obtained from BBBike.org, (b) the Chicago city boundaries in the form of GIS shape files were obtained from The City of Chicago (2023a), and (c) the PUMA CSV files and PUMS GIS shape files for Chicago were obtained from the US Census Bureau. The downloaded data sets were supplied to CITYSCAPE to generate a model for the city of Chicago. The model generation was conducted on an 8-core (i7-3770K CPU @ 3.50GHz, with 4 physical cores) machine with 16 GB of RAM running a stock Fedora Linux. CITYSCAPE was compiled using the GNU Compiler Collection (GCC) Version 8.3.1. Generating the model for Chicago took 130 seconds of runtime and required ~6.6 GB of RAM (average of 5 runs). On average 4.5 cores out of the possible 8 cores were utilized by CITYSCAPE. The average number of cores used is fewer than 8 because not all aspects of model generation are multithreaded. The corresponding single-threaded run took 395 seconds (average of 5 runs) suggesting a performance gain of 3×. The resulting model is shown in Figure 6 and has been color-coded based on the number of households assigned to a building. For example, in a large apartment complex in Chicago, 600 households were assigned to it. We have manually validated this assignment as discussed in Section 4.1.

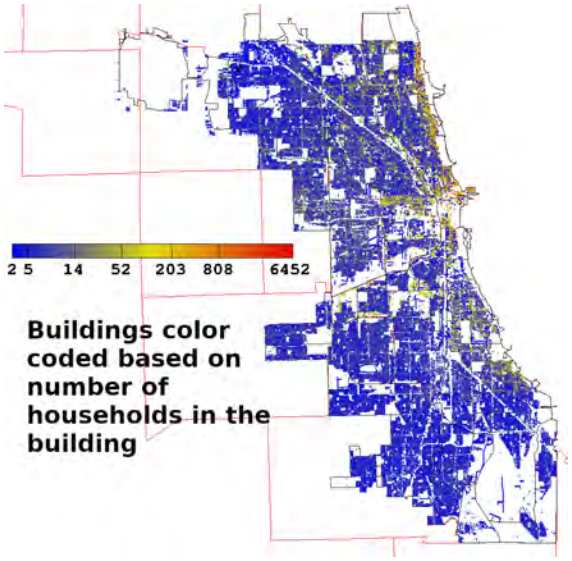


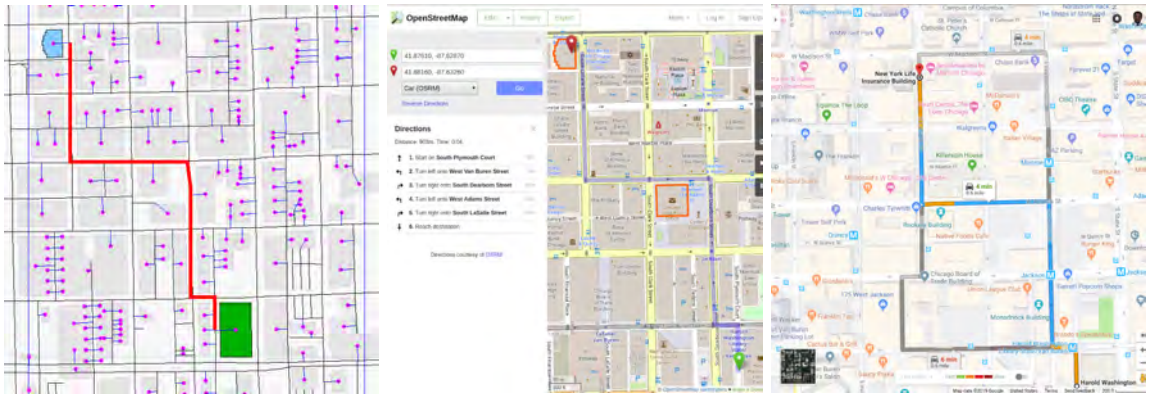
Figure 6: Chicago model generated by CITYSCAPE.

The model was generated using 1012 polygons from the different GIS shape files. From the OSM data, the model contained 217,654 nodes constituting 69,503 road segments. The model included 850,455 buildings of which 529,893 (~62.3%) were residential and 320,562 (~37.7%) non-residential buildings. Note that the OSM data only had 796,625 buildings and the other 53,830 residential buildings are synthetic homes generated by CITYSCAPE as discussed in Section 3.1. These residential buildings were generated on the 3,281 residential roads in OSM data set that did not have any houses on them. The PUMS data had a total of 1,109,393 households and the model included the same number of households. In the case of Chicago, the number of households in the generated model is fewer by ~4% (*i.e.,* 1.1 million instead of 1.11 million) and the average household size is lower by ~6% (*i.e.,* 2.27 persons per household instead of 2.41). These differences are due to the limitations of the PUMS data as discussed in Section 3.4.1.

4.1 Verification and Validation of the Chicago Model

The generated model contains two key characteristics that need to be verified and validated. The first feature is the layout of buildings and connecting roadways. The second characteristic is the assignment of households to buildings. Verification and validation of these characteristics in a large model is a complex task and exhaustive verification is impractical. Correspondingly, we have adopted different approaches. First, we verified and validated household assignments to buildings via statistical sampling. We randomly selected 50 households in the model and manually verified if the building assigned to them was meaningful using Google Maps. Building addresses were obtained using the OSM-IDs in the generated model. We have also manually verified assignments of households to large apartment complexes by sorting building populations and verifying the top 50 buildings. With this sample size, we achieve a statistical confidence level of 85% that the generated model contains a meaningful assignment of households to buildings.

Next, we proceeded to verify connectivity between buildings using the roadways in the model. For this verification, we have used a subset of Chicago Taxi Trips data from the The City of Chicago (2023a). The taxi trip data provides information about approximate pick-up and drop-off locations at the centroid of the 80 communities in Chicago shown in Figure 3. The times for the taxi rides are rounded to the nearest 15 minutes. For our verification, we randomly select two buildings in the start and destination areas and compute the fastest route between the two buildings and verify that the trip time is consistent with those recorded in the taxi trips. We have developed a custom A* pathfinding algorithm to compute the fastest taxi routes. We then manually compare the taxi routes with those generated by OpenStreet Maps and Google Maps to verify that the routes are consistent as summarized by the example in Figure 7. We have manually verified 50 routes and their timings, giving us a margin of error of <15%. Collectively our verification and validation provide us with 85% confidence that the generated model is a digital twin of Chicago, thereby validating our method for model generation.



(a) Route in CITYSCAPE (b) Route in OpenStreet Maps (b) Route in Google Maps

Figure 7: Example of comparing route and timings to validate road network in our generated model.

5 CONCLUSIONS AND FUTURE WORK

The need to analyze, design, and proactively implement sophisticated policies at city levels has rapidly grown to effectively manage ongoing challenges due to increased urbanization, emergent infectious diseases, and city planning. Simulation-based methods are gaining broad applicability in this area due to their advantages. However, simulations require valid, realistic models that effectively characterize human demographics. Several model generation methods have been proposed for generating either group-based (households or city-blocks) or individual-based models as discussed in Section 2. Each of these types of models has its respective advantages and disadvantages. In this work, we propose a novel, *ab initio* method for model generation and we have incorporated it into a software suite called CITYSCAPE. Our model includes both group-level characteristics in the form of households and individual characteristics. This permits simulation and analysis at two different scales. These model attributes are extracted from Public Use Micro Sample (PUMS) and Public Use Microsample Areas (PUMAs) data from the US Census Bureau. The modeler can optionally include over 100 different attributes for households and individuals in the model. In addition, the model includes detailed roadways and buildings. These model features make it amenable to hierarchical, group-based, or individual-based analyses.

In this paper, we have used the city of Chicago (IL, USA) as a case study due to several reasons as discussed in Section 4. Even though Chicago is a large city (population of ~2.69 million), CITYSCAPE was able to process all of the data and generate a model in a relatively short time of 130 seconds on a modest computer (an i7-3770K CPU @ 3.50GHz, with 4 physical cores) and used ~6.6 GB of RAM. Although we have not conducted scalability analyses, a rough estimate suggests that these are acceptable characteristics for processing larger cities such as Los Angeles (population ~3.77 million) or New York (population ~7.88 million). Moreover, modern computers can have 40 cores and nearly 1 TB of RAM which would provide more than ample resources for generating models of even larger cities such as Tokyo, New Delhi, or Shanghai.

We have conducted several experiments to verify and validate the generated model and transitively CITYSCAPE and our proposed model generation method. The model included all of the PUMS households and hence it was statistically indistinguishable from the census data. We have manually verified household-to-building assignments. In addition, we have used taxi trip data to validate the roadways generated in our model. Based on statistical sampling, our experiments provide an 85% confidence that the generated model of Chicago is a realistic digital twin. The model can be used for further analyses, including simulation-based analyses of a broad range of applications. The potential sources of errors in the model mainly stem from the approximate nature of PUMS data. This error may translate to errors in certain types of analyses conducted using the model. However, the extent of the errors in simulations or analyses can only be quantified once such analyses are pursued. We are in the process of developing a simulation framework that utilizes the generated model for the analysis of traffic accidents and sustainable transportation. These applications will provide a more concrete use case of the generated model and its effectiveness.

5.1 Future Work

This paper presented the current outcomes from our ongoing efforts to develop digital twins of cities. We are continuing to extend our method to include additional features. For example, we are planning to include group quarters PUMS records so as to include people living in group quarters such as hospitals, hotels, etc. Currently, the models include both individuals and households. It is feasible to further abstract the model to a larger area and such abstractions are planned for future work. Our CITYSCAPE software is freely available on https://github.com/raodj/cityscape. Nevertheless, we are planning to develop a convenient website where modelers can create custom models for their analysis (Rao, Chernyakhovsky, and Wilsey 2000). We are also exploring the application of our model for simulation-based analysis by extending our parallel simulator called MUSE (Higiro, Gebre, and Rao 2017) with application to computational epidemiology, sustainable transportation, and smart city planning.

REFERENCES

Balcan, D., B. Gonçalves, H. Hu, J. J. Ramasco, V. Colizza, and A. Vespignani. 2010, August. "Modeling the Spatial Spread of Infectious Diseases: The GLobal Epidemic and Mobility Computational Model". *Journal of Computational Science* 1(3):132–145.

Barrett, C. L., R. J. Beckman, M. Khan, V. A. Kumar, M. V. Marathe, P. E. Stretz, T. Dutta, and B. Lewis. 2009. "Generation and Analysis of Large Synthetic Social Contact Networks". In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, edited by M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, 1003–1014: IEEE.

Bhatele, A., J.-S. Yeom, N. Jain, C. J. Kuhlman, Y. Livna, K. R. Bisset, L. V. Kale, and M. V. Marathe. 2017, May. "Massively Parallel Simulations of Spread of Infectious Diseases over Realistic Social Networks". In *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, 689–694.

Chen, X., and F. B. Zhan. 2008, Jan. "Agent-based Modelling and Simulation of Urban Evacuation: Relative Effectiveness of Simultaneous and Staged Evacuation Strategies". *Journal of the Operational Research Society* 59(1):25–33.

CIESIN 2016. "Gridded Population of the World, Version 4 (GPWv4): Population Count". NASA Socioeconomic Data and Applications Center (SEDAC). Center for International Earth Science Information Network, Columbia University.

Daniels, R. S. 2007, September. "Revitalizing Emergency Management after Katrina". *Public Manager* 36:16–20.

Giridharan, N., and D. M. Rao. 2016, July. "Eliciting Characteristics of H5N1 in High-Risk Regions Using Phylogeography and Phylodynamic Simulations". *Computing in Science Engineering* 18(4):11–24.

Higiro, J., M. Gebre, and D. M. Rao. 2017. "Multi-Tier Priority Queues and 2-Tier Ladder Queue for Managing Pending Events in Sequential and Optimistic Parallel Simulations". In *Proceedings of the 2017 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*, SIGSIM-PADS '17, 3–14. New York, NY, USA: Association for Computing Machinery.

Jones, D., C. Snider, A. Nassehi, J. Yon, and B. Hicks. 2020. "Characterising the Digital Twin: A Systematic Literature Review". *CIRP Journal of Manufacturing Science and Technology* 29:36–52.

Keeling, M. J. 2005. "Models of Foot-and-mouth Disease". In *Proceedings of the Royal Society B: Biological Sciences*, Number 1569, 1195–1202.

Longini, I. M., A. Nizam, S. Xu, K. Ungchusak, W. Hanshaoworakul, D. A. T. Cummunings, and M. E. Halloran. 2005. "Containing Pandemic Influenza at the Source". *Science* 309(5737):1083–1087.

Oak Ridge National Laboratory (ORNL) 2012, Oct. "LandScan™Geographic Information Science and Technology (GIST)".

OpenStreetMap contributors 2023. "Planet Dump Retrieved from https://planet.osm.org". https://www.openstreetmap.org.

Rao, D. M., A. Chernyakhovsky, and V. Rao. 2009. "Modeling and Analysis of Global Epidemiology of Avian Influenza". *Environmental Modelling & Software* 24(1):124–134.

Rao, D. M., V. Chernyakhovsky, and P. A. Wilsey. 2000, January. "WESE: A Web-based Environment for Systems Engineering". In *2000 International Conference On Web-Based Modelling & Simulation (WebSim'2000)*. Society for Computer Simulation.

Rao, M. E., and D. M. Rao. 2021. "The Mental Health of High School Students During the COVID-19 Pandemic". *Frontiers in Education* 6.

Schmidt, E., and D. M. Rao. 2017, October. "Generating Synthetic Individual Human Population and Activity Models". In *Proceedings of the 31st European Modelling and Simulation Conference (ESM'17)*, 127–134.

The City of Chicago 2023a, Apr. "Chicago Data Portal".

The City of Chicago 2023b. "Chicago Data Portal: Boundaries - Wards (2015–2023)". https://data.cityofchicago.org/Facilities-Geographic-Boundaries/Boundaries-Wards-2015-2023-/sp34-6z76.

United States Census Bureau 2023a. "Public Use Microdata Areas (PUMAs)". https://www.census.gov/programs-surveys/geography/guidance/geo-areas/pumas.html.

United States Census Bureau 2023b. "Public Use Microdata Sample (PUMS)". https://www.census.gov/programs-surveys/acs/microdata.html.

Unwin, H. J. T., S. Mishra, V. C. Bradley, A. Gandy, T. A. Mellan, H. Coupland, J. Ish-Horowicz, and et al. 2020. "State-level Tracking of COVID-19 in the United States". *Nature Communications* 11(6189).

Verity, R., L. C. Okell, I. Dorigatti, P. Winskill, C. Whittaker, N. Imai, G. Cuomo-Dannenburg, and et al. 2020. "Estimates of the Severity of Coronavirus Disease 2019: A Model-based Analysis". *The Lancet Infectious Diseases* 20(6):669–677.

Wang, Y., L. Shi, J. Que, Q. Lu, L. Liu, Z. Lu, Y. Xu, J. Liu, Y. Sun, S. Meng, K. Yuan, M. Ran, L. Lu, Y. Bao, and J. Shi. 2021. "The Impact of Quarantine on Mental Health Status Among General Population in China During the COVID-19 Pandemic". *Molecular Psychiatry* 26:4813–4822.

AUTHOR BIOGRAPHIES

DHANANJAI (DJ) M. RAO is an Associate Professor in the Department of Computer Science and Software Engineering. His primary research interest is in the area of modeling, parallel simulation, and High-Performance Computing (HPC) with application to epidemiology, viral phylodynamics, genetic analyses, ecology, and recently smart cities and transportation. His email address is raodm@miamiOH.edu and his lab page is https://pc2lab.cec.miamioh.edu/.