

ORIGINAL ARTICLE

Open Access



Automated design of gradual zone systems

An iterative algorithm to design optimally sized spatial zones suitable for spatial modeling, while respecting municipal boundaries

Joseph Molloy^{1*}  and Rolf Moeckel²

Abstract

Background: The appropriate resolution of a zone system is key to the development of any transport model, as well as other spatial analyses. The number and shape of zones directly impacts the effectiveness of any further modeling steps, with the trade-off between computation time and model accuracy being a particularly important consideration. Currently, zone systems are often designed by hand. The gradual rasterization zoning algorithm produces good empirical results by computationally generating raster cells of varying area, but similar population and employment.

Methods: We address several limitations of the original algorithm in this paper. Firstly, the allocation of employment to raster cells is weighted by land use instead of by area percentage. Secondly, the algorithm is extended to respect municipal delineations. Raster cells are split along these delineations, with any undesirably small zones created by this process re-merged with neighbors in the same municipality. Aligning the generated cells to municipalities simplifies and improves the disaggregation of socioeconomic data. An iterative algorithm has been developed to automatically determine the threshold that results in the zone system of the desired size. Raster cells are split along these delineations, with any undesirably small zones created by this process re-merged with neighbors in the same municipality.

Results and Conclusion: Aligning the generated cells to municipalities simplifies and improves the disaggregation of socioeconomic data. Using this algorithm, a zone system has been generated for the Munich metropolitan region. Only 13 iterations were needed to converge within 5% of target of 5000 delineated zones. The improved algorithm maintains the advantages of the original algorithm and adds several important improvements that are useful when creating a zone system.

Keywords: Zone system, Quadtree Algorithm, Modifiable areal unit problem, Spatial modeling, Traffic analysis zones

Background

The spatial resolution of geographic data has a very big impact on the results of any spatial analyses. Eidlin [10] for example, showed that New York City is the densest place in the U.S. if city boundaries are chosen as the unit of analysis, whereas the selection by metropolitan areas, in contrast, would assign this title to Los Angeles. This issue was labelled by Openshaw [20] as the Modifiable Area Unit Problem (MAUP), expressing that results of spatial

analyses are influenced by the chosen zone size. Similarly, changing the zone system in transport modeling requires complete recalibration of the model. Zones should be small enough to reduce the number of intrazonal trips [7, 9], but large enough to minimise the number of zones and keep model runtimes short. Spiekermann and Wegner [23] note that the selection of the appropriate zone size is contextual, while others describe it as more art than technique [13, 16]. More often than not, zone systems were defined by local authorities decades ago and are adopted for any spatial analysis [14, 21]. Changing a zone system or creating a new one is very labor intensive, and hence, rarely done [8].

*Correspondence: joseph.molloy@ivt.baug.ethz.ch

¹ETH Zürich IVT - Institute for Transport Planning and Systems, Stefano-Franscini-Platz 5 8093, Zürich, Switzerland

Full list of author information is available at the end of the article

Whenever a zone system is created from scratch, it traditionally requires manual labor to decide which neighborhoods shall form one zone, which streets, rivers or other geographical features should act as boundaries between zones, and how large zones shall be. Being a manual process, it is extremely unlikely that the relative spatial resolution is proportional in different parts of the study area. Other approaches have used uniform raster cells to cover the entire study area [5, 6]. While regular raster cells are very efficient for certain spatial analysis, including Cellular Automata, they are inefficient for transport modeling and other computing-intensive analyses dealing with complicated zonal interactions [4, 5]. In contrast, zones should be larger where there is less activity and smaller where there is more activity. This way, most resources are allocated to areas that deserve most attention by the analyst.

In some cases, zones have been defined by individual land parcels [22, 24, 26]. While parcel-level data are very useful for many visualizations, no transport or land use model for larger metropolitan study areas is known to be fully operational. The sheer number of parcels makes it impossible to efficiently run complex simulations.

In an attempt to overcome these issues, Moeckel and Donnelly [18] created a tool to generate an automated zone system for the Georgia statewide model. The tool applied the quadtree algorithm and repeatedly subdivided larger raster cells into four smaller raster cells, until each raster cell had a population of no more than 5000 households. The tool created smaller raster cells in urban areas and larger raster cells in rural areas, and it was used successfully for a transport model. However, raster cells ignored jurisdictional boundaries, such as city, county and state boundaries. This shortcoming made it impossible to correctly allocate socio-demographic data, which was always given by jurisdiction, to raster cells. Furthermore, the algorithm led to some raster cells dominated by large bodies of water, or with no population at all, adding unnecessary computational requirements to further analyses. While the quadtree algorithm is still applied at the core of the research presented here, new features have been developed to address these shortcomings. Land use is now considered when disaggregating population and employment, and municipal boundaries are respected to aid in data disaggregation and the development of hierarchical models. Finally, an automated approach is implemented to identify the required threshold parameter for the algorithm.

Approaches to zoning are often proprietary and vary widely. With each project using different manual methods or techniques that have been developed in house. This work presents a step forward towards a more open approach to zoning analysis. The algorithm is simple to understand and highly automated. Yet it is configurable

and can be applied using only Open Data and no commercial libraries or software such as ArcGIS. Our methods are designed to be used with whatever data the analyst has available. As a general requirement, Open Data needs to respect the privacy of individuals, and are often aggregated to the municipal level or higher. As such, zoning systems that adhere to administrative boundaries will aid the analyst in working with Open Data. While the data used in this analysis are not Open Data, it is worth noting that the openness of municipal data such as population, employment, and land use varies around the world, and is at a minimum often available without charge from relevant statistical authorities for non-commercial use. The code is open source and available at https://github.com/msmobility/silo_zoneSystem.

The paper proceeds as followed. “Literature review” section covers the MAUP problem, its implications for zone system design, and reviews the previously developed method. “Methods” section describes the algorithm and its features. “Application” section presents the creation of a zone system for the metropolitan area of Munich using our approach. “Discussion” section discusses the benefits and drawbacks of the method, and “Conclusion” section concludes.

Literature review

The significant variety of approaches in the literature towards zone system design is a tribute to the complexity and importance of the process. Moeckel and Donnelly [18] note that the zone systems used for spatial analyses and spatial modelling have different requirements. While analysis requires only that zones accurately represent statistical data in a spatial sense, zone systems for modeling also need to avoid zones of odd shapes, such as donuts and horseshoes. Openshaw [19] proposed the automatic zoning procedure (AZP), a rule based approach to iteratively aggregate smaller zones to best fit certain statistical measures. Eventually, this approach was computerized using GIS software, extending its applicability to thousands of zones [21].

Another automated approach to update existing zone systems was constructed by Cockings et al. [8], which split zones with increasing population, and merged those where the population was declining. Batty [3] developed a procedure that defines a zone system to maximize social entropy. Based on the concept of entropy from thermodynamics, spatial entropy is defined as the distribution of spatial data over an area in such a way that the information content cannot be increased.

Such automated spatial analysis zone systems are not suitable for spatial modeling due to the irregular-shaped zones they produce. The zone shape is particularly important in transportation models, as trip origins and destinations are calculated using the zone’s centroid. In some

cases, such as donuts and horseshoe-shaped zones, the centroid may lie outside the zone area. Hence, zone systems need to be specifically designed for spatial modeling.

In a similar vein to the identification the MAUP problem in spatial analyses [20], multiple studies have showed the impact of zone system design on spatial modeling results [7, 15, 25]. Viegas, Martinez, and Silva [25] investigated MAUP in spatial modelling by analyzing the impact on intrazonal trips and zero-trip zones of various zone system resolutions. Lovelace, Ballas and Watson [15] investigated commute trips and confirmed that smaller zone sizes improved the fit of the model to observed data. These studies suggest that zone systems should be tailored to specific use cases in spatial modeling. However, typically this is not the case, primarily due the time and cost required to revise existing zone systems and repopulate them with socioeconomic data.

A particularly interesting approach was presented by Hagen-Zanker and Jin [12], called adaptive zoning. For every origin zone, destination zones are aggregated together based on their distance from the origin. Hence, a separate map is created for each origin, with nearby destination zones being small, and more distant ones larger. They tested the method on a commuting model in England and found the results were equivalent to the conventional model, despite a reduction in the number of zone pairs by 96% and computation time by 70%.

The introduction of computer systems made the use of raster cells attractive in spatial modelling. They are homogeneously shaped, easy to process geometrically, and have simple relationships to their the adjacent cells. Approaches using cellular automata to model land-use [5] and urban growth [1] have represented locations using raster cells and their interactions with adjacent neighbours. Moeckel [17] also used raster cells to create and compare land use models using firms versus those using employees.

Approaches using raster cells present some key challenges. Firstly, socio-economic data needs to be accurately disaggregated to these raster cells. Spiekermann and Wegner [23] presented one solution. As part of methodology for disaggregating zone systems, they generated probabilities of population and employment for each raster cell based on land-use data available at the size of the smallest raster cell. Monte-Carlo sampling with these probabilities was then used to allocate socio-economic data to these raster cells.

In the creation of an Origin-Destination matrix for transport modeling, each cell not only needs to interact with its adjacent neighbours, but all other zones as well. If the number of cells needed to cover a study area at the necessary resolution in a raster cell zone system is very large, the number of interactions between non-adjacent cells make the model computationally infeasible.

Moeckel and Donnelly [18] proposed a gradual rasterization method to retain the benefits of raster cells, while reducing the number of zones. Smaller cells are created in dense metropolitan areas, and larger cells in rural areas. In doing so, they were able to programmatically define a raster cell zone system suitable for transport modeling.

Previously developed method

The gradual rasterization method to create a zoning system was first proposed by Moeckel and Donnelly [18] to model traffic along the I-75 corridor in Georgia. The GDOT (Georgia Department of Transportation) statewide model [2] was used to analyze transportation improvements along the I-75 from Atlanta, Georgia to Chattanooga, Tennessee. It was found that along the section of the I-75 within the Atlanta Metropolitan region, travel demand was substantially overestimated by the model. Further investigation showed that an increase in geographical resolution within Atlanta improved results, suggesting that a higher spatial resolution in urban areas was needed. To do this, the authors proposed their gradual rasterization method to improve spatial detail in denser areas while avoiding an exponential increase in the size of the trip table.

The study area was rasterized into the smallest raster cells to be considered. A square covering Georgia was rasterized into 4096 x 4096 raster cells. The number of cells must be a power of two for the quadtree algorithm to work. Population data was then disaggregated to this raster. Population and employment were allocated proportionally to each cell based on the area percentage of the various intersecting zones.

The quadtree algorithm created the gradual raster cells. The algorithm started with one large cell covering the entire study area. If the summed population and employment of this cell exceeded the specified threshold, the cell was divided into 4 cells of equal size. This was recursively repeated for the new cells until the population of each cell was below the threshold, or the cell was of the minimum raster cell size. In this way, the number of zones was reduced by having many smaller cells in areas of higher population, and fewer larger cells elsewhere. Moeckel and Donnelly based this decision on a rule proposed by Flowerdew, Feng and Manley [11], that zones across a study area should have a similar number of households. The threshold had to be specified manually.

Moeckel and Donnelly's approach noteworthy improved the model results. Through trial and error it was found that a threshold of 5000 units of population and employment resulted in suitable zone system consisting of almost 5000 zones. They found it remarkable that the overall model validation was improved only through changes to the spatial resolution of the assignment step, without modifications to the model design. The gradual

rasterization kept roughly the same number of zones in rural areas, where the GDOT model performed well, but added zones to areas where the GDOT model under performed in urban areas. They noted that while this process could have been performed manually, it would have risked introducing inconsistencies in the spatial resolution. A straight forward, instead of gradual, rasterization to the grid of the smallest cell size would have resulted in 4 million cells. With so many raster cells, the creation of trip tables and their assignment would have become infeasible.

Objectives

Moeckel and Donnelly’s algorithm has some limitations listed below. In the following sections of this paper solutions to these shortcomings are proposed.

1. Raster cells can overlap multiple jurisdictions, resulting in a ‘secondary’ zone system that is not nested within the original set of zones or municipalities. This lack of hierarchy introduces added complexity and errors when assigning socioeconomic data or trip ends to raster cells.
2. Population and employment are distributed to raster cells by the area percentage of the overlapping zones. This unrealistically assumes that socioeconomic data such as population and employment are evenly distributed throughout the zones or municipalities.
3. The process of identifying a population and employment threshold that results in the desired spatial resolution and number of zones was a manual process of trial and error.
4. Every zone that exceeds a threshold value is split into four cells of equal size. If population was only present in one corner of this zone, three out of four newly created raster cells would have no population. Thereby, resources are allocated inefficiently to some degree.

Methods

The algorithm described is composed of three main parts. 1) The allocation of socioeconomic data to minimum sized raster cells. 2) the quadtree algorithm. 3) the split and merging of cells along municipal boundaries. Steps 2 and 3 are repeated in the automated iterative search for a suitable zonal resolution. The quadtree algorithm is unchanged from the previous work and its description can be found in the literature review above.

Allocation by landuse

In the first step of the algorithm, a raster is defined at the size of the smallest desired zone size, i.e. 100x100m. Regional statistics must be allocated to these raster cells, with the parent municipality of each cell determined from the cell centroid. Population and employment are allocated individually to each cell based on the land use.

$$x_i = X_m \cdot \frac{\sum_K (f_k \cdot l_i^k)}{\sum_{j \in m} \sum_K (f_k \cdot l_j^k)}$$

where

- x_i = value for cell i
- X_m = value for municipality m
- l_i^k = area of land-use of type k in cell i
- f_k = scaling factor for land-use k

Population is distributed based on the percentage area of a raster cell of residential and mixed-use land-use, and employment is allocated by the weighted area percentage of various land uses such as mixed-use and industrial. Figure 1 shows how the weighting by land-use provides a much more realistic distribution of employment.



Fig. 1 Employment in the city of Munich and surrounding suburbs - Jobs per raster cell. **a** Municipal Employment equally distributed among raster cells. **b** Disaggregation by land use

Split and merge

While pure square raster cells have significant computational advantages, they rarely match up to the irregular metropolitan boundaries defined over centuries by geography and politics. To improve the disaggregation of socioeconomic data, it is desirable to create raster cells whose borders align with municipal boundaries. The delineation to municipal boundaries forms the bulk of the improvements to this algorithm. It is performed as a two step process, which we call split and merge. The quadtree rasterization algorithm remains unchanged from the original work by Moeckel and Donnelly, only translated to the programming language Python.

For each municipal region, the intersecting raster cells are identified. These zones are replaced with two or more new cells that represent the intersection of the raster cell and municipality. If the result of the intersection is a multipolygon, a new cell is created for each sub-polygon. For each new segment of the original cell, the population and employment is recalculated. The algorithm then determines if the cell is acceptable, or if it must be merged with a neighbor in the same region.

The cells to be merged are then compared with all adjacent cells in the same region for which their combined value would not exceed the threshold, and the neighbor with the longest shared boundary is selected. This process is repeated until that cell can no longer be merged without exceeding the threshold. When no suitable neighbor can be found to merge with, the cell is included in the final output.

Procedure 1 Split raster cells

Input: *regions, raster_cells*
created_cells = set()
for each region in regions do
 region_cells \leftarrow *intersection(region, raster_cells)*
 for each cell in region_cells do
 if region does not contain cell then
 for each newcell in intersection(cell, region) do
 raster_cells.add(newcell)
 merge_set.add(newcell)
 region_cells.remove(cell)
 merge(region_cells, merge_set)
 created_cells.add(region_cells)
Output: *created_cells*

The neighbour search criteria is refined to prevent the occasional creation of 'donut' shaped zones (see Fig. 2). An additional predicate is added, specifying that any potential merged cell must be a simple polygon, with the centroid of its polygon lying within the merged area. These

Procedure 2 Merge raster cells

Input: *region_cells, merge_set*
while *merge_set* is not empty **do**
 cell = any cell from *merge_set*
 if $|region_cells| = 0$ **and** $|merge_set| = 1$ **then**
 region_cells.add(cell)
 return *region_cells*
 else
 neighbor = *find_best_neighbour(cell, region_cells)*
 if suitable neighbor **then**
 merged_cell = *union(neighbor, cell)*
 region_cells.remove(cell)
 region_cells.remove(neighbor)
 merge_set.add(merged_cell)
 else
 region_cells.add(cell)

two conditions, while not necessarily important for land-use analysis, are essential in transport modeling for the calculation of both inter- and intra-zone travel times.

Finally, the results of the split and merge for each municipality are recombined to form the resulting zone system. In this way, the algorithm respects municipal boundaries, maintains regular zone shapes, and avoids the creation of small or under-populated zones.

Automating trial & error

Moeckel and Donnelly [18] observed a monotonically decreasing power curve relationship between the population threshold and the number of zones. Taking advantage of this fact, a binary search is performed along the solution space to iteratively identify the threshold, which results in the desired number of zones.

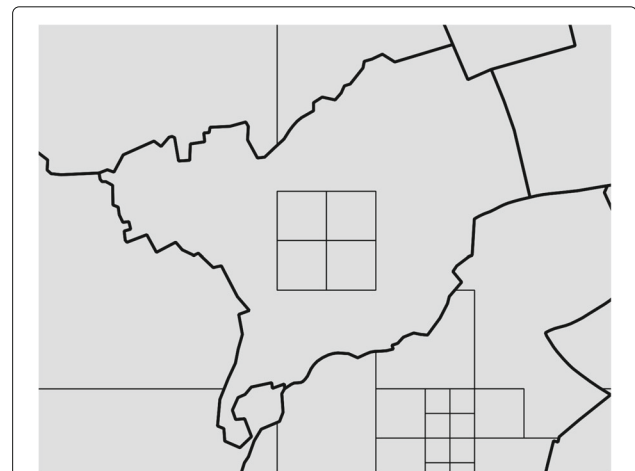


Fig. 2 Example of a donut shaped zone avoided with an additional predicate

First, the bounds of the search area are defined,

Let x_{max} = the total population and employment

Let $x_{min} = 1$

Let $x = \frac{x_{max} + x_{min}}{2}$

The algorithm is then run for the threshold x and the number of zones returned. If the number of zones is within the specified percentage of the threshold, a suitable zoning system has been found. If it is too high, then the lower threshold x_{min} is set to x . If it is too low, then the upper threshold x_{max} is set to x . x is then recalculated as the average of the new minimum and maximum bounds, and a new zone system generated. This process is repeated until a solution is found.

Application

We applied our approach to the Munich metropolitan region to generate a zone system for a larger integrated land use/transportation model in the region. The region comprises the city of Munich, nearby municipalities and any satellite cities such as Ausburg that share a strong commuting link with Munich. In total, the region consists of 444 municipalities, with a total population of nearly 4.5 million, and total employment of 1.8 million. 29% of the population live in the city of Munich itself, while the average population per municipality is only less than 10,000 persons. It is clear that a higher spatial resolution is needed for the city of Munich than for other less populated municipalities. In our study area, there are also two large lakes, the Starnbergersee and the Ammersee, which are popular destinations for leisure travel. These geographical obstacles present a good test for our zoning algorithm.

For the study area, the following data were available:

- Shapefile of the municipalities comprising the study area.
- Population raster data at 100x100m resolution from the 2011 German census.
- Population and employment for each municipality in the study area for the year 2008.
- Land-use vector data covering the study area at a parcel level.

Since employment data were not available for 2011 to match the census population data, the socioeconomic data from 2008 was used. Due to the large variation in residential density throughout the Munich region, the 2011 census raster was used to disaggregate population to raster cells, as it provided a better allocation than our land use method, and the data were available. Where such

a high resolution raster of population density is not available, the disaggregation by land use still provides acceptable results. Employment was disaggregated by land use, with the weightings of each category presented in Table 1.

Results

The minimum input raster cells were created at a 100x100m resolution. The iterative algorithm was run with a goal of creating 5000 zones within a tolerance of 5%. The converged solution used a population and employment threshold of 2295 to produce 4902 zones. Only 13 iterations were needed to converge on a result. The result was achieved on a single core 2.7 GHz processor in just over 8 min, with each iteration taking on average 40 s. The number of iterations needed can be reduced to 3 or 4 by specifying a sensible initial upper bound for the threshold such as 10,000. In Fig. 3, the results of each iteration are plotted against the run time. The algorithm makes increasingly larger steps through the search space, before it overestimates the required number of zones and backtracks to reach the target range. Naturally, the testing of higher thresholds takes longer as more detailed zone systems result. Hence, the iteration starts with higher thresholds, and moves quickly through the initial search space before testing lower thresholds that generate more higher resolution zone systems.

Resulting zone system

The resulting zone system is presented in Fig. 4. As desired, smaller cells are created in urban areas, while larger cells cover regional areas. Many municipalities are now covered by only a few cells, which nicely divide the municipalities. Smaller cells are still retained for population centers of both urban and regional municipalities. The zones also respect the boundaries of the major lakes and forest areas.

The three stages of the algorithm are portrayed in Fig. 5. In step (a), the quadtree algorithm is applied, generating the square raster cells. (b) shows the results of the split process, which generates many low-populated cells along municipal boundaries. In (c), the cells with little population are merged with larger raster cells in the municipality. By merging cells along the longest boundary, the created cells still have a relatively regular shape.

Table 1 Land-use weightings for allocation of employment data

Category	Weighting
Residential	0.0
Special	0.1
Mixed Use	0.4
Industry	0.5
Sport+Rec	0.0

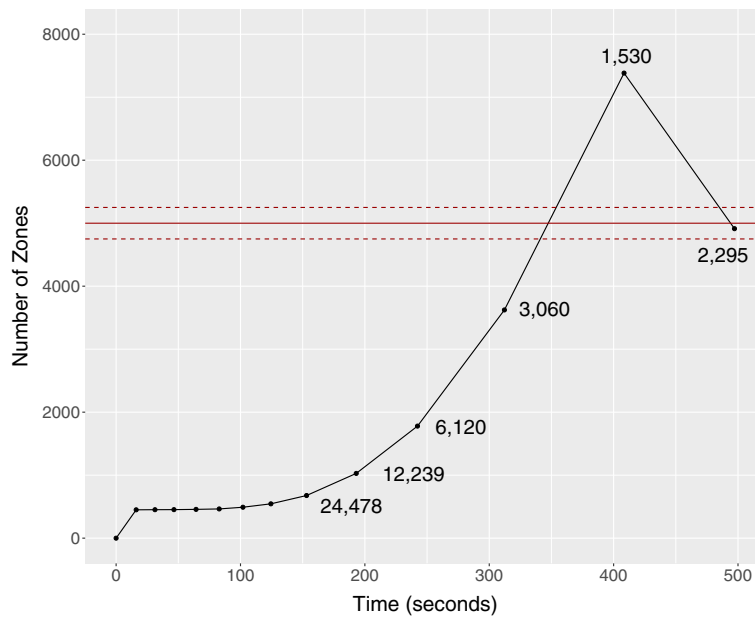


Fig. 3 Convergence of the iteration process to generate 5000 zones +/- 5%. Points labeled with the threshold used in that step

Discussion

To the best of the authors’ knowledge, there are no standard ways to statistically prove the generalized benefits of a particular zoning approach. Depending on the application, zone systems have different requirements. As an alternative, the statistical characteristics of the resulting zone system are compared to the intermediate results produced by the original quadtree algorithm.

In Fig. 6, the distribution of the population by raster cell shows how the distribution changes through the split and merge process in the Munich scenario. Following the quadtree step, the distribution of the population and employment among cells is relatively good. However, in splitting the cells that lie along municipal boundaries, the number of zones increases by 62% due to the number of municipal boundaries present. It is clear from Fig. 6 that many of these new cells are near-empty. Here, the importance of the merge step is evident, as it combines these ‘slivers’ with other cells within the same municipality, reducing the number of low populated raster cells when compared to the split stage by 55.4%. The end effect is an improvement over the original quadtree algorithm, as the density is better distributed across the value range and the total number of zones is lower. The split and merge process achieves a more balanced distribution of population and employment across zones than the quadtree algorithm.

As with the quadtree approach presented by [18], zone size is still inversely proportional to urban densities, creating smaller zones where there is more activity. Secondly,

zones are nested within municipal boundaries, making it easier to use census data. To avoid a detrimental increase in the number of insignificant zones along boundaries, the number of zones is reduced by merging small snippets with neighboring zones. Finally, the algorithm is flexible

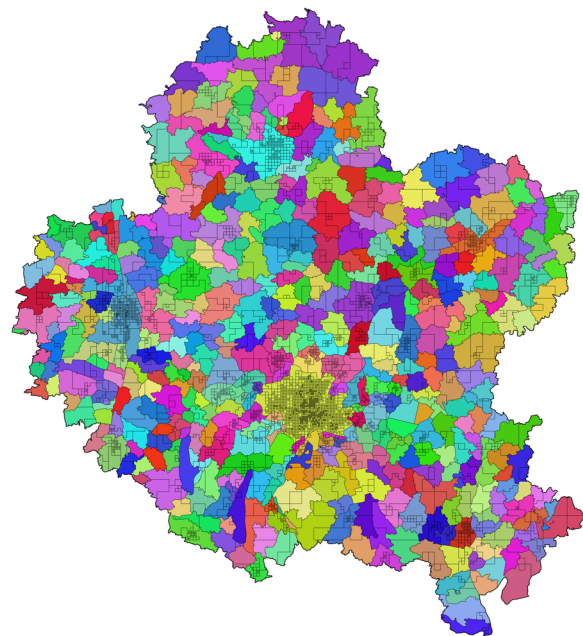
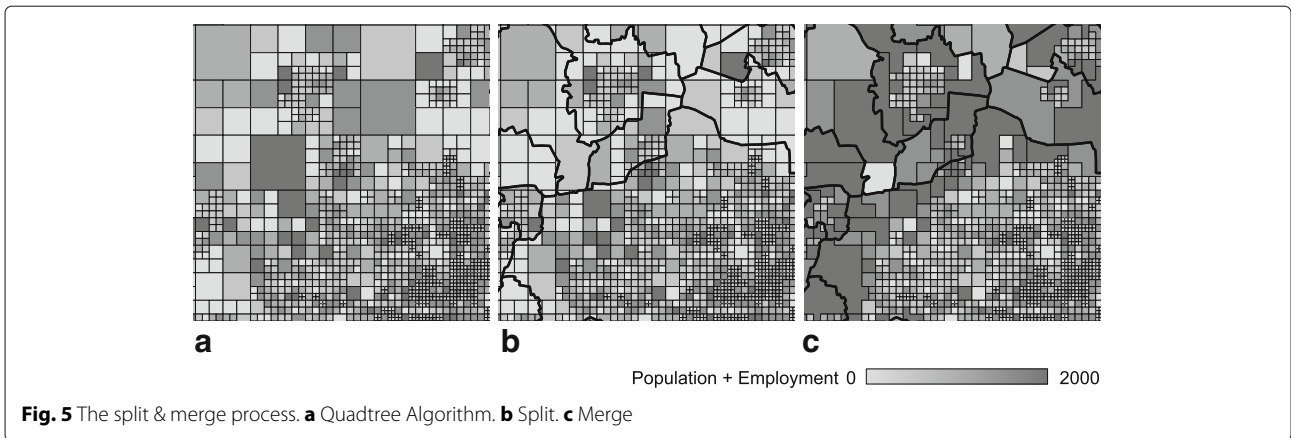


Fig. 4 The resulting zone system for the Munich metropolitan area, developed using the approach described in this paper. Zones are colored by their municipal parent, to show how the boundaries are respected. The smaller zones represent areas of higher density, with rural areas covered by larger zones

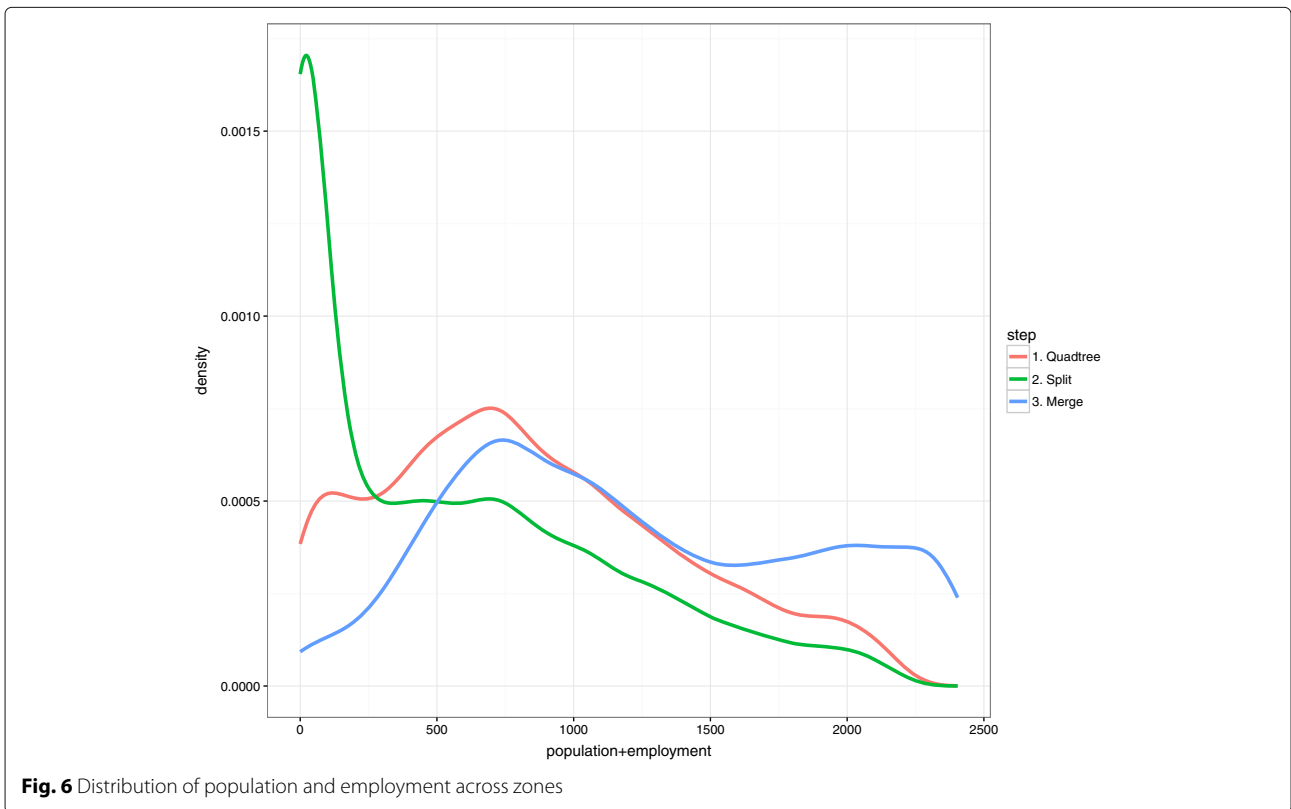


and allows the user to calibrate the number of zones for a particular purpose.

The automation of the trial-and-error process also provides significant work-flow advantages to the analyst. The algorithm was used to create a very reasonable zone system for the Munich metropolitan region without manual trial-and-error. Hence, when integrated into a larger transport model, the zone system can be easily updated to meet project requirements.

Our algorithm does have some disadvantages. In splitting and merging raster cells along municipal boundaries,

the homogeneous square geometry across all raster cells is forfeited, and any generated zone system should still be checked for reasonableness by the analyst. Hence, for some applications, such as cellular automata, the boundary respecting algorithm described in this paper will not be appropriate. Improvements in spatial resolution and zonal accuracy nearly always come at a cost. It is up to the analyst to make the trade off between model simplicity and the need to accurately allocate socioeconomic data from aggregated data sources. While the algorithm does produce only simple polygonal zones, i.e. no donuts,



it does not guarantee the non-existence of other oddly shaped zones.

Conclusion

In conclusion, our improved algorithm maintains the advantages of the original algorithm and adds several important improvements that are useful when creating a zone system. Foremost, through respecting municipal boundaries, it creates a zone system that nests the created raster cells within the municipal regions (or any other higher-ranking geography defined by the user). This allows for easier and more accurate allocation of socioeconomic data, which is often only available at the municipal level.

This is achieved while retaining the gradual spatial resolution of the original quadtree algorithm, providing more spatial detail in areas of high density to better allocate population and employment. The split and merge procedures work in tandem to further improve the distribution of population and employment, reducing the number of lowly populated zones in regional areas and along municipal boundaries when compared to the original quadtree approach.

The algorithm is also able to take into account lakes and other geographical boundaries which need to be considered in the zone system. Having a better measure of the actual land occupied by a zone will improve the disaggregation and assignment of data such as trip-ends to cells.

Some applications, such as cellular automata, require regular shaped zones, and as such, the boundary respecting algorithm described in this paper will not be appropriate for such applications. In other areas, such as transport modeling, the added benefits when it comes to the allocation of data only available at the municipal level should be obvious to transport modelers.

More empirical work still needs to be done to evaluate the model performance of our new algorithm. To this end, the suitability of the created zone system for the Munich metropolitan region will be assessed through the current work developing an integrated land use and transport model for the region. Further plans to implement this work in Cape Town, Dublin and Melbourne will appraise the adaptability of the algorithm to other study areas and modeling scales.

Acknowledgments

The research was completed with the support of the Technical University of Munich - Institute for Advanced Study, funded by the German excellence initiative and the European Union seventh framework programme under grant agreement n. 291763.

A presentation based on this work was presented at the 2016 ESRI Developers Summit, Berlin by the corresponding author.

Authors' contributions

The idea for the work was conceived by RM. RM and JM jointly produced the literature review and collected the data. The methodology was designed and

applied by JM. The body of the article was drafted by JM, with critical revision and final approval provided by RM. Both authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹ETH Zürich IVT - Institute for Transport Planning and Systems, Stefano-Franscini-Platz 5 8093, Zürich, Switzerland. ²Modeling Spatial Mobility, Department of Civil, Geo and Environmental Engineering, Technical University of Munich, Arcisstr. 21 80333 Munich, Germany.

Received: 14 March 2017 Accepted: 13 June 2017

Published online: 24 July 2017

References

- Andersson C, Lindgren K, Rasmussen S, White R. Urban growth simulation from first principles. *Phys Rev E*. 2002;66(2):026–204.
- Atkins. Development of statewide model report. model v2.0 draft report. Tech. rep. Atlanta, GA: Georgia Department of Transport; 2011.
- Batty M. Spatial entropy. *Geogr Anal*. 1974;6(1):1–31.
- Batty M. Cellular automata and urban form: a primer. *J Am Plann Assoc*. 1997;63(2):266–74.
- Batty M, Xie Y, Sun Z. Modeling urban dynamics through gis-based cellular automata. *Comput Environ Urban Syst*. 1999;23(3):205–33.
- Berling-Wolff S, Wu J. Modeling urban landscape dynamics: A review. *Ecol Res*. 2004;19:119–29.
- Chang Kt, Khatib Z, Ou Y. Effects of zoning structure and network detail on traffic demand modeling. *Environ Plan B Plan Des*. 2002;29(1):37–52.
- Cockings S, Harfoot A, Martin D, Hornby D. Maintaining existing zoning systems using automated zone-design techniques: methods for creating the 2011 census output geographies for england and wales. *Environ Plan A*. 2011;43(10):2399–418.
- Ding C. The gis-based human-interactive taz design algorithm: examining the impacts of data aggregation on transportation-planning analysis. *Environ Plan B Plan Des*. 1998;25(4):601–16.
- Eidlin E. What density doesn't tell us about sprawl. *Access Mag*. 2010;1(37):2–9.
- Flowerdew R, Feng Z, Manley D. Constructing data zones for scottish neighbourhood statistics. *Comput Environ Urban Syst*. 2007;31(1):76–90.
- Hagen-Zanker A, Jin Y. A new method of adaptive zoning for spatial interaction models. *Geogr Anal*. 2012;44(4):281–301.
- Hargrove W, Hoffman F. Potential of multivariate quantitative methods for delineation and visualization of ecoregions. *Environ Manag*. 2004;34(1):S39–60.
- Haynes R, Daras K, Reading R, Jones A. Modifiable neighbourhood units, zone design and residents' perceptions. *Health Place*. 2007;13(4):812–25.
- Lovelace R, Ballas D, Watson M. A spatial microsimulation approach for the analysis of commuter patterns: from individual to regional levels. *J Transp Geogr*. 2014;34:282–96.
- Martínez LM, Viegas JM, Silva EA. A traffic analysis zone definition: a new methodology and algorithm. *Transportation*. 2009;36(5):581–99.
- Moeckel R. Firm location choice versus job location choice in microscopic simulation models. In: *Employment Location in Cities and Regions*. Berlin, Heidelberg: Springer; 2013. p. 223–42.
- Moeckel R, Donnelly R. Gradual rasterization: redefining spatial resolution in transport modelling. *Environ Plan B: Plan Des*. 2015;42(5):888–903.
- Openshaw S. A geographical solution to scale and aggregation problems in region-building, partitioning and spatial modelling. *Trans Inst Br Geogr*. 1977;2:459–72.
- Openshaw SS. The modifiable areal unit problem. In: *CATMOG 38*, GeoBooks. Norwich; 1984.
- Openshaw S, Rao L. Algorithms for reengineering 1991 census geography. *Environ Plann A*. 1995;27(3):425–46.
- Pendyala RM, Konduri KC, Chiu YC, Hickman M, Noh H, Waddell P, Wang L, You D, Gardner B. An integrated land use - transport model

system with dynamic time-dependent activity-travel microsimulation. *Transport Res Record J Transport Res Board.* 2303;2013:19–27.

23. Spiekermann K, Wegener M. In: Fotheringham AS, Wegener M, editors. *Freedom from the tyranny of zones: towards new gis-based models.* London: Taylor and Francis; 2000, pp. 45–61.
24. Venter CJ, Lamprecht T, Badenhorst W. Demographic and regional economic modeling using stochastic allocation in the city of johannesburg. In: *Transportation Research Board (TRB).* 2006. p. 14.
25. Viegas JM, Martínez LM, Silva EA. Effects of the modifiable areal unit problem on the delineation of traffic analysis zones. *Environ Plan B: Plan Des.* 2009;36(4):625–43.
26. Waddell P, Moore T, Edwards S. Exploiting Parcel-Level GIS for Land Use Modeling. In: *ACSE Conference. Transportation, Land Use, and Air Quality: Making the Connection;* 1998. p. 10.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ springeropen.com
