

The max- p -compact-regions problem

Xin Feng¹ | Sergio Rey² | Ran Wei²

¹Department of Geography and Environmental Sustainability, The University of Oklahoma, Norman, Oklahoma 73019, USA

²School of Public Policy, Center for Geospatial Sciences, University of California, Riverside, California, USA

Correspondence

Xin Feng, Department of Geography and Environmental Sustainability, The University of Oklahoma, Norman, OK 73019, USA.
Email: selenafeng@ou.edu

Funding information

National Science Foundation, Grant/Award Number: 1831615

Abstract

The max- p -compact-regions problem involves the aggregation of a set of small areas into an unknown maximum number (p) of compact, homogeneous, and spatially contiguous regions such that a regional attribute value is higher than a predefined threshold. The max- p -compact-regions problem is an extension of the max- p -regions problem accounting for compactness. The max- p -regions model has been widely used to define study regions in many application cases since it allows users to specify criteria and then to identify a regionalization scheme. However, the max- p -regions model does not consider compactness even though compactness is usually a desirable goal in regionalization, implying ideal accessibility and apparent homogeneity. This article discusses how to integrate a compactness measure into the max- p regionalization process by constructing a multiobjective optimization model that maximizes the number of regions while optimizing the compactness of identified regions. An efficient heuristic algorithm is developed to address the computational intensity of the max- p -compact-regions problem so that it can be applied to large-scale practical regionalization problems. This new algorithm will be implemented in the open-source Python Spatial Analysis Library. One hypothetical and one practical application of the max- p -compact-regions problem are introduced to demonstrate the effectiveness and efficiency of the proposed algorithm.

1 | INTRODUCTION

Regionalization has been an important area of research in GIScience, involving aggregating larger numbers of smaller areas into a smaller set of larger regions to fulfill a particular goal. For example, endogenous neighborhoods can be identified by clustering neighboring small-scale geographic units to executive functions such as detecting neighborhood growth, change, inequality, and clustering types, in human development and global sustainability (e.g., Rey et al., 2011; Spielman & Singleton, 2015). For another example, in precision agriculture, management zones are delineated by aggregating field units that share similar soil physical and chemical characteristics (Li, Shi, Li, & Li, 2007). Actual nutrient needs are furnished specifically in each management zone to enhance the quality and productivity of crops. More application cases can be easily found in many well-known social and environmental contexts, such as sale territory assignment, school districting, political districting, legislative districting, natural resource management, and zone aggregation for economic modeling (e.g., Hess & Samuels, 1971; Hess, Weaver, Siegfeldt, Whelan, & Zitlau, 1965; Li, Church, & Goodchild, 2014; Niemi, Grofman, Carlucci, & Hofeller, 1990). In these analyses, regions can be determined using spatial aggregation models.

Although many regionalization models and algorithms have been developed over more than 50 years, one significant challenge raised in the past decade is the definition of the number of regions to be designed. For most regionalization models and algorithms, it is always necessary to specify the number of regions in advance, and this specification will directly affect the regionalization results. However, many practitioners do not know ahead of time how many regions should be created when they aggregate areas. Duque, Anselin, and Rey (2012) developed a clustering problem called the max- p -regions problem to endogenize the number of regions. The essence of the max- p -regions problem is to set up a condition every region must satisfy and then let the data dictate. The condition is usually based on a spatially extensive attribute, such as area per region, population per region, and travel flow per region, and thus the regions are designed to be suitable for the following analysis.

Besides avoiding subjectivity in the definition of scale (number of regions), the max- p -regions problem imposes no constraints on the shape of the regions. This characteristic may be reasonable in some applications, while the consideration of compactness is essential in others. For example, it is desirable to have a compact police patrol area to ensure an efficient response, because the travel time from one point to another in a compact area can be minimized to the greatest extent (Camacho-Collados, Liberatore, & Angulo, 2015). Having relatively compact electoral districts is one possible way to minimize political gerrymandering (Niemi et al., 1990). During the Covid-19 pandemic, the development of a compact residential area can significantly reduce trips to essential destinations such as grocery stores, pharmacies, and transit stations (Hamidi & Zandiatashbar, 2021). In these relevant examples, a compactness central model is meaningful and necessary. Actually, the shape of the region has been well studied as a specific requirement in empirical applications when the number of regions is prespecified. Take the p -compact-regions problem as an example, which involves aggregating a large set of spatial units into a pre-defined number p of compact and contiguous regions (Li et al., 2014). So, what about the unknown number of regions? How to formulate and solve a regionalization model that cares about the shape of the region but does not predefine the number of regions? How does the shape requirement affect the unknown maximum number of identified regions? These are the main issues to be solved in this paper.

This article mathematically formalizes a max- p -compact-regions problem to generate the maximum number of compact regions from a large set of spatial units. The next section provides a review of the compactness measure in regionalization and recent solution techniques of regionalization models. This is followed by an extension of the max- p -regions problem that accounts for the compactness of aggregated regions. A heuristic solution approach involving four main stages—shape processing, region growing, enclave assignment, and local search—is then derived. Results for two compactness-driven applications are presented to highlight the influence of compactness in the max- p -regions problem, the capabilities of the proposed max- p -compact region problem, and the efficiency of its associated algorithms.

2 | BACKGROUND

The family of regionalization methods can be divided into two categories: supervised regionalization and unsupervised regionalization. Supervised regionalization methods always assume prior knowledge about the aggregation process, including relevant variables for the aggregation, a prespecified number of regions, spatial contiguity constraints within each region, and a particular aggregation criterion to be optimized (Duque, Ramos, & Suriñach, 2007; Govorov, Becony  , Gienko, & Putrenko, 2019). For example, the p -regions model (Duque, Church, & Middleton, 2011) and its extensions have been developed to aggregate a large set of unit areas into p geographically connected regions. Similarly, the constrained clustering methods such as SKATER and REDCAP prespecify the number of edges to be cut in a continuous tree to group spatial units into contiguous regions (Assun  o, Neves, C  mara, & da Costa Freitas, 2006; Guo, 2008; Guo & Wang, 2011). Their associated objective functions vary, such as minimizing within-region dissimilarity (Duque et al., 2011), maximizing between-region equality (Weaver & Hess, 1963), and maximizing region compactness (Li et al., 2014).

Compared to these supervised methods, unsupervised regionalization methods usually dispense with certain prior knowledge settings. For example, in the field of land use/land cover (LULC), an unsupervised regionalization model helps identify and delineate a certain number of landscape pattern types by dividing a large area into spatial units containing unique LULC patterns (Niesterowicz, Stepinski, & Jasiewicz, 2016). The procedure does not require spatial contiguity constraints either implicitly or explicitly, while still predefining the number of landscape pattern types.

For most of the existing supervised and unsupervised regionalization methods, the number of regions is always prespecified, even though it is rarely known in advance and its determination is complex. There are only a few exceptions, one of which is the max- p -regions problem (Duque et al., 2011; Ghiggi, Puliafito, & Zoppoli, 1975; Oehrlein & Haurert, 2017). In the max- p -regions problem, how many regions need to be designed is unknown, but we can identify the criteria that each region must satisfy. Some specific requirements are allowed such as within-region spatial contiguity and threshold for the aggregation, based on which the number and the shape of regions are automatically dictated by the data. The original max- p -regions problem has been extended in recent years from different perspectives. Folch and Spielman (2014) emphasized the strength of the original max- p -regions algorithm to provide more flexibility in the constraints used to define the characteristics of a region. She, Duque, and Ye (2017) incorporated network structure into the max- p -regions problem, enabling a balance between the corridor regions formation and regional heterogeneity. Wei, Rey, and Knaap (2021) improved the heuristic approach of the max- p -regions problem by combining simulated annealing and tabu search to reach high accuracy and computational efficiency. In this article, we will extend the max- p -regions problem by integrating compactness as an explicit goal.

Compactness has been an essential requirement in many regional science applications since the 1960s. Compactness implies high accessibility, so it has always been required in many applications with a need for efficiency and equality of response/service (e.g., Fryer & Holden, 2011; Hess et al., 1965; Niemi et al., 1990). Ensuring the regions are as compact as possible is also an indirect way to pursue spatial contiguity constraints (Duque et al., 2007; Weaver & Hess, 1963). Besides, the maximization of regional compactness is likely to increase the within-region similarity in attributes and properties. For example, compact sales territories tend to have a common community of interest which may result in higher sales (Hess & Samuels, 1971). Therefore, compactness has been demonstrated to be a useful characteristic, but it is not defined in the max- p -regions problem. Our proposed model aims to fill this gap. Therefore, compactness has been implicitly or explicitly introduced into optimization models in regionalization (Duque et al., 2007). Indirect measurements, such as minimizing the distance from an area to a given center and minimizing the perimeter lengths of regions, were embedded into the objective function of a mixed integer programming problem (Li, Goodchild, & Church, 2013). Explicit measurements include the ratio of the perimeter to the shape area, the ratio of the area of a region to the minimum bounding area of a given shape, and the moment of inertia (Datta, Malczewski, & Figueira, 2012). Direct measurements are usually more accurate

in measuring the shape of an area in comparison to indirect measurements. A detailed review of compactness measurements and their integration in optimization models can be found in Duque et al. (2007) and Li et al. (2013).

The computation of compactness described above has been applied to regionalization models in which the number of regions is prespecified. In this article we explore how to consider compactness when p is unknown, or more specifically, how to incorporate the calculation of a compactness measurement into the max- p -regions problem. However, solving the problem efficiently is a big challenge. The max- p -regions problem is NP-hard and computationally expensive to solve (Duque et al., 2012). The exact mixed-integer programming (MIP) solution method can only handle a very small problem (e.g., 16 units in Duque et al., 2012), even using the latest commercial MIP solver, GUROBI (Wei et al., 2021). The corresponding heuristic method has been developed by Duque et al. (2012), making it computationally possible to solve actual scaled problems. Wei et al. (2021) further improved the method and enabled the max- p -regions problem to address large-scale neighborhood delineation. However, the integration of compactness calculations makes the original max- p -regions even more challenging to solve. Indirect compactness measurements are more straightforward to calculate, but their values are unstable for a given shape when the size and spatial resolution change. Direct compactness measurements can provide a more accurate description of the region's shape; however, they usually make the optimization problem nonlinear and it becomes more challenging to obtain optimal solutions (Li et al., 2014). Therefore, advances in effectively and efficiently solving the max- p -regions problem while considering compactness are needed.

3 | MODEL FORMALIZATION

To begin, assume that compactness is neglected. Formulation of the max- p -regions problem relies on the following notation (Duque et al., 2012):

- i, l = index and set of areas, $l = \{1, \dots, n\}$
- k = index of potential regions, $k = \{1, \dots, n\}$
- c = index of contiguity order, $c = \{0, \dots, q\}$, with $q = n - 1$
- d_{ij} = dissimilarity relationships between areas i and j , with $i, j \in l$ and $i < j$
- I_i = spatially extensive attribute value of area i , with $i \in l$
- T = minimum value (threshold) for attribute I at the regional scale

$$w_{ij} = \begin{cases} 1, & \text{if areas } i \text{ and } j \text{ share a border, with } i, j \in l \text{ and } i \neq j, \\ 0, & \text{otherwise} \end{cases}$$

$$N_i = \{j | w_{ij} = 1\}$$

$$h = 1 + \left\lceil \log \left(\sum_i \sum_{j \in N_i} d_{ij} \right) \right\rceil$$

We also have two decision variables:

$$y_{ij} = \begin{cases} 1, & \text{if areas } i \text{ and } j \text{ belong to the same region,} \\ 0, & \text{otherwise} \end{cases}$$

$$x_i^{kc} = \begin{cases} 1, & \text{if area } i \text{ is assigned to region } k \text{ in order } c, \\ 0, & \text{otherwise} \end{cases}$$

We do not know a priori the number of regions that will be created in the max- p -regions problem, therefore, the “potential regions”, indexed by k , could range from 1 to the total number of areas. When region k starts to grow from its “root” area i , x_i^{k0} equals 1, which means that area i is assigned to region k in order 0. With the index of contiguity order c , an ordering system is used in this model to ensure contiguity within one region. The non-root areas are assigned to a region only if the area is adjacent to an area assigned to the same region with a smaller order number. The size criterion that each region must satisfy is based on the spatially extensive attribute l_i . One region can finish growing only if the total attribute value of the areas within this region exceeds the threshold. The max- p -regions problem is as follows:

$$\text{minimize} \left(- \sum_{k=1}^n \sum_{i=1}^n x_i^{k0} \right) \times 10^h + \sum_i \sum_{j|j \sim i} d_{ij} y_{ij} \quad (1)$$

subject to

$$\sum_{i=1}^n x_i^{k0} \leq 1, \forall k \quad (2)$$

$$\sum_{k=1}^n \sum_{c=0}^q x_i^{kc} = 1, \forall i \quad (3)$$

$$x_i^{kc} \leq \sum_{j \in N_i} x_j^{k(c-1)}, \forall i, k, c \quad (4)$$

$$\sum_{i=1}^n \sum_{c=0}^q x_i^{kc} l_i \geq T \sum_{i=1}^n x_i^{k0}, \forall k \quad (5)$$

$$y_{ij} \geq \sum_c x_i^{kc} + \sum_c x_j^{kc} - 1, \forall i, k, c \quad (6)$$

$$y_{ij} \in \{0, 1\}, \forall i, j \quad (7)$$

$$x_i^{kc} \in \{0, 1\}, \forall i, k, c \quad (8)$$

Equation (1) consists of two terms. The first seeks a maximum number of areas designated as “root,” which is equivalent to maximizing the number of regions, p , since each region has one and only one “root” area. This term is multiplied by -1 because the objective function is formulated as a minimization problem. The second term aims to minimize the total pairwise dissimilarities between areas assigned to the same region. These two terms are merged by introducing an implicit hierarchy so that the number of regions, p , reaches the maximum first, and then reducing the heterogeneity within the p regions. To achieve this hierarchy, a scaling factor h is multiplied by the first term. By doing so, a feasible solution with a larger value of p will always be preferred over any other solution with a smaller value of p no matter what their amount of heterogeneity is. Heterogeneity is only compared between solutions with the same number of regions, within which lower heterogeneity will be preferred.

Constraint 2 ensures each region does not have more than one “root” area. Constraint 3 specifies that each area must be assigned to precisely one region with one contiguity order. Constraint 4 restricts the assignment

made for an area i to region k in order c only if there exists an area j , adjacent to i , that is assigned to the same region k in order $c - 1$. Constraint 5 ensures that the sum of the spatially extensive attribute in one region will exceed the prespecified threshold. Constraint 6 builds the connection between two types of decision variables, ensuring the binary variable y_{ij} equals 1 whenever areas i and j are assigned to the same region k . Constraints 7 and 8 indicate binary integer restrictions on decision variables.

Clearly, the max- p -regions model only considers region contiguity but does not include any concern for region compactness. To encourage region compactness, we extend the max- p -regions model into a new optimization model, the max- p -compact-regions model, where a compactness measure is integrated into the objective functions. Specifically, the max- p -compact-regions involves two objectives that aim to maximize the number of regions as well as optimize the overall compactness of these regions, and it is formulated as follows:

$$\text{maximize } \sum_{k=1}^n \sum_{i=1}^n x_i^{k0} \times 10^{h'} + \sum_{k=1}^n C(Z_k) \quad (9)$$

where $C(Z_k)$ represents the compactness of region k and Z_k is the set of areas that are assigned to this region. Similar to the objective function of the max- p -regions problem, a scaling factor $h' = 1 + \lfloor \log(\sum_{k=1}^n C(Z_k)) \rfloor$ is used here to ensure that a solution having a larger value of p is always preferred to any other solution with a smaller p no matter how compact the regions are. The overall compactness of the regions is optimized only when the maximum p is reached. By weighting the two objectives with the scaling factor, the model is forced to compare total compactness between solutions with the same number of regions. The proposed max- p -compact-regions model does not involve the within-region dissimilarity component as the max- p -regions model. However, the dissimilarity component can be added into the model formulation and solving process, which will be briefly illustrated in Section 6.

Research related to the development of an effective compactness measure has been of interest within the scientific community. These measures can be grouped into four categories: the area-perimeter approach, shape-reference approaches, geometric pixel measurement, and the elements dispersion approach (Li et al., 2013). In this article, we adopt a direct compactness measure based on a normalized moment of inertia (NMI) because of its accurate description, proven effectiveness and outstanding computational efficiency (Li et al., 2013, 2014). The compactness index $C(Z_k)$ for region k is defined by:

$$C(Z_k) = \frac{A_{Z_k}^2}{2\pi I_{Z_k}^G} \quad (10)$$

where A_{Z_k} is the area of region k containing a set of areas Z_k , and $I_{Z_k}^G$ is the second moment of inertia (MI) of the region about an axis perpendicular to it and passing through its centroid G . The MI can be calculated as a region grows when a neighboring basic unit is aggregated (or indeed as a region shrinks when a unit is removed). That is, the MI of a new region can be derived by summing the MI of the origin region before the aggregation and the MI of the neighboring unit that will be aggregated into the region. More details of the calculation process are given in Li et al. (2014). $C(Z_k)$ ranges from 0 to 1, where 1 refers to the most compact shape (a circle), and 0 refers to an infinitely extending shape. The compactness measurement based on Equation (10) is clearly nonlinear; therefore, the max- p -compact-regions problem, formulated in Equations (2)–(9), is a nonlinear integer programming problem.

4 | SOLUTION APPROACH

Given the known expensive computation in solving the max- p -regions problem and the nonlinear nature of the integrated compactness measure, a heuristic solution approach is developed to solve the max- p -compact-regions problem efficiently. The approach consists of four main stages: shape processing, region growing, enclave

assignment, and local search. The first stage calculates the compactness for each unit area; the second stage focuses on growing regions by adding neighboring units in a certain way to maximize the number of regions; the third stage assigns enclaves, which are unassigned unit areas left from the second step, to neighboring regions; and the final stage is to iteratively improve the overall compactness of regions through a customized simulated annealing that integrates a tabu list.

Figure 1 shows the flowchart of the proposed solution approach for the max- p -compact-regions problem, in which the context associated with the introduction of compactness is highlighted in red. After initialization, the shape processing phase works on precomputing the MI of each unit area following its definition in Li et al. (2013). Due to the large randomness in the procedure, the region growing phase will lead to different partitioning results. Therefore, we repeat the procedure a certain number of times and only pass the partition results having the maximal number of regions to the subsequent two phases of enclave assignment and local search. Finally, the solution determined by this approach is the partition with the best overall compactness. The details of the three stages of region growing, enclave assignment, and local search are as follows.

The region growing phase focuses on identifying as many regions as possible that satisfy the threshold. It starts with randomly selecting an unassigned unit as the seed unit of a region. Then the unassigned neighbors of the units in the region will be added into the region until the total spatial extension attribute of the region reaches the threshold. In this process, the joining order of adjacent units is related to their ability to improve the current region's compactness. One unit from the top N units that optimize the compactness of the current region is randomly selected to add to the region. When there are not enough unassigned neighboring units to add, the region may fail to reach the threshold. If so, all units assigned to this region are called "enclaves" and are moved to the enclave set for processing in the next phase. This phase is ended by repeating the process until all units are assigned to a region or included in the enclave set. Therefore, a set of contiguous regions and a set of enclaves are identified as the output of this phase.

The enclave assignment phase aims to allocate enclaves to regions that have been identified in the region growing phase. Its first step is to randomly select a unit from the enclave set and randomly assign it to one of the neighboring regions with N best compactness values. This process is repeated until all enclaves have been assigned to a region. The enclave assignment phase ends with a set of contiguous regions consisting of all of the units.

Having obtained an initial feasible solution through the first three phases, the goal of the local search phase is to further improve the overall compactness of the solution. In general, it is achieved by iteratively moving spatial units from their current regions to neighboring regions and then checking the possibility of improving the total compactness of regions. Duque et al. (2012) tested the performance of simulated annealing and tabu search for local search to minimize the total within-region heterogeneity. According to their findings, the tabu search is more likely to capture the best solution, while simulated annealing is more computationally efficient. Wei et al. (2021) combined the two algorithms' advantages to solve the max- p -regions problem, achieving better-quality solutions and higher-efficiency computations.

The local search algorithm for max- p -compact-regions is an extension of the algorithm in Wei et al. (2021), and it is also designed with the integration of simulated annealing and a tabu list. However, the goal is to optimize the overall compactness of regions as formulated in Equation (9). The process is described in Pseudocode 1. Specifically, given a feasible solution, all candidate units which can move to a neighboring region without violating the contiguity and threshold constraints are identified and added into a potential unit (PU) list. Then we randomly select one candidate unit from the list and identify its move to the neighboring region with the best compactness change, which is referred to as the recipient region. If the move can improve the total compactness, accept the move and add its reverse move to a tabu list. Otherwise, the non-improving move might also be accepted with a certain probability if it is not in the tabu list. The probability is defined by Boltzmann's equation, $p = e^{-\Delta C/t}$, where ΔC is the total compactness change because of the move and t is the current temperature. The temperature is initialized to be 1 and gradually decreases at a prespecified cooling rate during

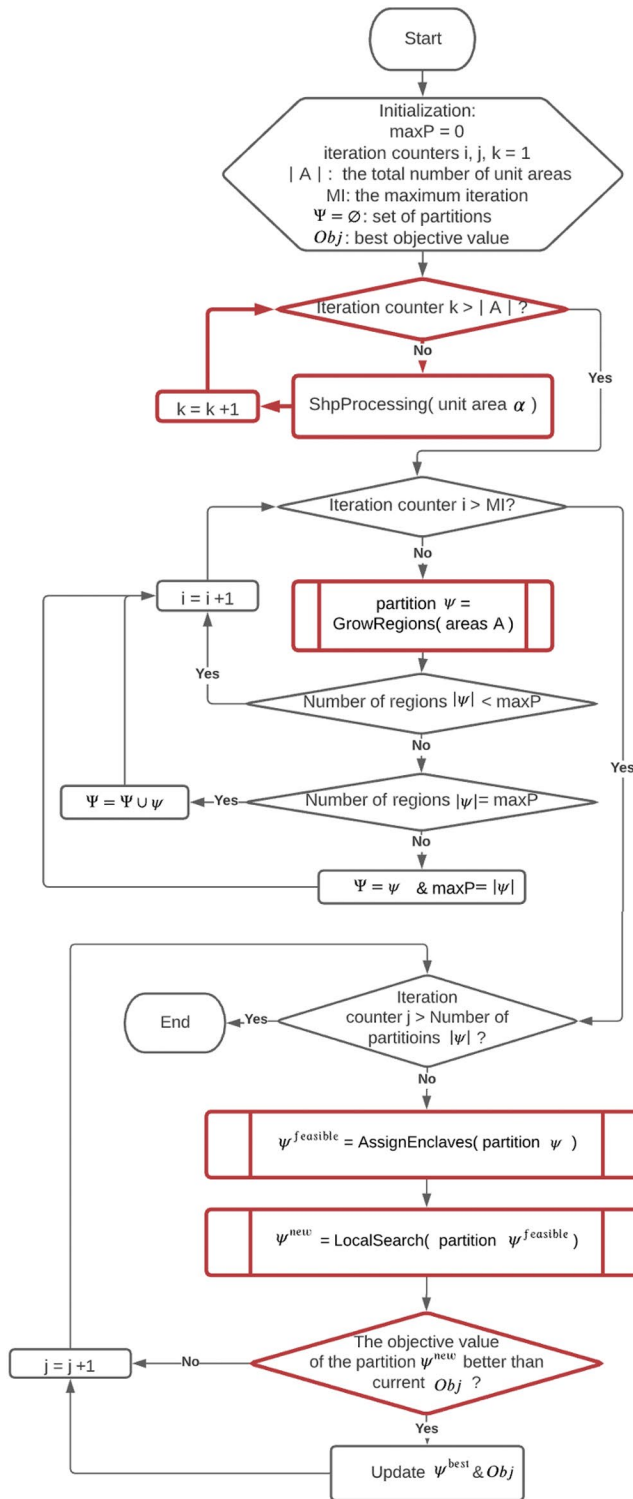


FIGURE 1 Flowchart of the proposed solution approach

the process. The tabu list (TL) is a list of banned moves to prevent the search from returning to previously visited solutions. The introduction of the tabu list has been demonstrated to eliminate search oscillation, thereby speeding up convergence. In sum, in the local search algorithm, the candidate units will be checked one by one for possible improvement and then removed from the potential unit list. A new potential unit list will be identified if the current one is empty. The phase will end once the total number of non-improving moves (NM) reaches the prespecified maximum non-improving moves value (MNM) or the temperature t is less than the minimum allowed (MA).

Pseudocode 1: Local search (NM, MNM, TL, PU, flag, t , MA, alpha)

```

while NM < MNM and  $t$  > MA do
    if |PU| = 0 then
        Identify PU
    else
         $i \leftarrow \text{SelectUnit (PU)}$ 
         $j \leftarrow \text{BestNeighbor (i)}$ 
        DonorRegion  $\leftarrow \text{GetRegion (i)}$ 
        RecipientRegion  $\leftarrow \text{GetRegion (j)}$ 
        if ImproveCompactness ( $i, j$ ) is True then
            NM  $\leftarrow$  0
            TL  $\leftarrow$  TL.add ( $j, i$ )
            MakeMove ( $i, j$ )
            flag  $\leftarrow$  1
        else
            NM  $\leftarrow$  NM + 1
            If Boltzmann's condition is satisfied then
                if ( $i, j$ ) in TL then
                    flag  $\leftarrow$  0
                else
                    MakeMove ( $i, j$ )
                    flag  $\leftarrow$  1
            else
                flag  $\leftarrow$  0
        PU.remove ( $i$ )
        if flag = 1 then
            PU.remove (any  $k$  in DonorRegion or RecipientRegion)
        else
             $t \leftarrow t * \text{alpha}$ 

```

5 | APPLICATIONS

In this section a series of applications are introduced to assess the ability of the proposed solution approach to solve the max- p -compact-regions problem. First, a hypothetical problem having a known optimal solution is used to demonstrate that the proposed solution approach can achieve optimality. Then experiments based on both hypothetical and practical data sets are conducted to explore the influence of the introduction of compactness to solve the max- p -compact-regions problem, specifically from the perspective of finding max p and compactness values and computation efficiency. Problem-solving was implemented in Python and ran on an Intel Core i7 CPU (3.2 GHz) with 32 GB of RAM.

The hypothetical data set is composed of 168 equilateral triangles of the same size as shown in Figure 2. These triangles are connected by vertices, and adjacent triangles share two vertices and an edge. The spatial extensive attribute value for each triangle is 1, and the threshold value is 24. The objective of this max- p -compact-regions problem is to aggregate the 168 triangles into the maximum number of compact and spatially contiguous regions such that the total value of the regional attribute is no less than 24. The number of regions is up to 7 ($=168/24$) in order to satisfy the threshold constraint. When the study area is partitioned into 7 regions with 24 basic units in each region, the most compact regions are regular hexagons, which has been demonstrated in Li et al. (2014).



FIGURE 2 A hypothetical study area for a max- p -compact-regions problem

Therefore, the optimal solution of this max- p -compact-regions problem is seven regular hexagons, each of them consisting of 24 triangles. Among the 1,000 times the solution approach executed for this problem, the percentage finding the true max- p value (7 in this case) and compactness index (.99 using the MI-based measure for a regular hexagon) was 100%. That is, all of the cases reach the optimal solution, as shown in Figure 3. The computation time for each time in this case is approximately 0.01 s. This experiment demonstrates that the proposed solution approach can obtain global optimal solutions for a max- p -compact-regions problem with a known optimal solution.

The practical application is based on a realistic data set consisting of 111,670 traffic analysis zones (TAZs) in six southern California counties (Ventura, Los Angeles, Orange, San Bernardino, Riverside, and Imperial); see Figure 4. The spatial extensive attribute value is the block-based population, which is interpolated in each TAZ by *pysal/Tobler* (<https://pysal.org/tobler/#>) (an open-source sublibrary for areal interpolation and dasymetric mapping of Python Spatial Analysis Library (PySAL)). Three different threshold values, $T = 5,000, 50,000$, and $100,000$, are tested.

The identified regionalization results corresponding to the threshold value of population 100,000 are presented in Figure 5a. In this case, 111,670 areas are aggregated into 126 relatively compact regions, and the best average compactness measurement is equal to .84. To clarify the difference between max- p -compact-regions and max- p -regions, the max- p -regions algorithm is also applied to this data set using the same threshold constraint. As shown in Figure 5b, only 119 regions are identified, with worse compactness.

An interesting thing we notice is that in this case the max- p value identified by max- p -compact-regions in Figure 5a is higher than that identified by the max- p -regions in Figure 5b under the same condition. The max- p -regions problem and the max- p -compact-regions problem do have different objectives: one considers dissimilarity

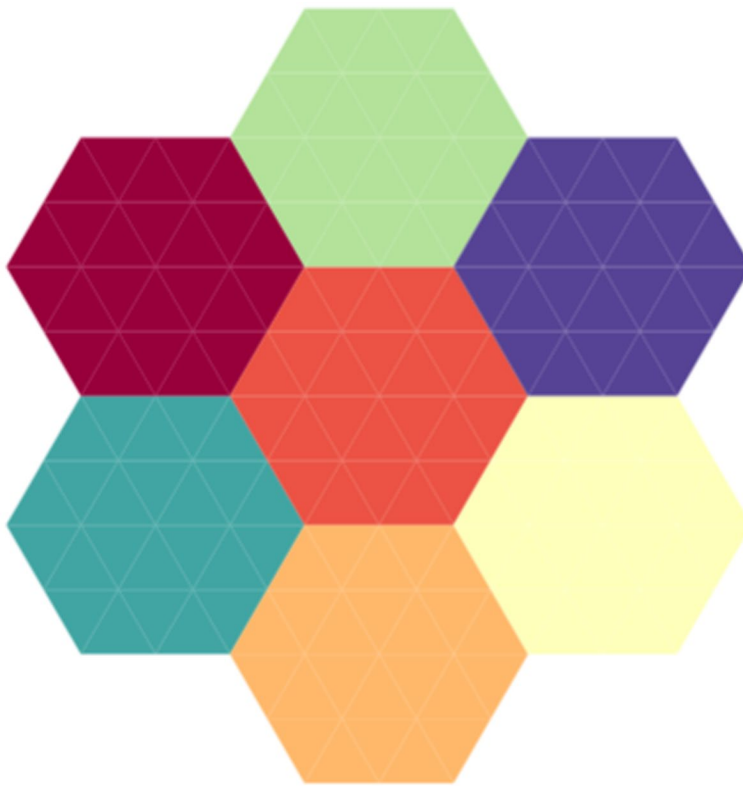


FIGURE 3 Partition of the optimal solution

and the other cares about compactness. However, their top priority is the same, to generate as many regions as possible. Whatever the amount of the second objective item is, a feasible solution with a larger value of p will always be favored over any other solution with a smaller value p for both problems. Therefore, if the solution of max- p -compact-regions problem can achieve a larger p , its partition can also be a good feasible solution for the corresponding max- p -regions problem.

For the max- p -regions problem, the region growing algorithm proposed in Wei et al. (2021) (which is also the one used in Figure 5b) devised the strategy that a region grows by iteratively adding neighboring unit areas to the region. They indicate their approach generally dominates the strategy in Duque et al. (2012) that always adds the top candidate that minimizes the total within-class dissimilarity in terms of the number of regions identified. Based on these findings, accounting for the attribute dissimilarity in the phase of region growing is not conducive to maximizing the number of regions for the max- p -regions problem.

The next interesting question is, for the max- p -compact-regions problems, whether the consideration of compactness affects the number of regions identified. Since the number of regions is determined during the region growing phase, the region growing algorithm is executed 1,000 times for each threshold to compare the number of regions identified with and without compactness concern. When considering compactness, regions are grown by iteratively adding a neighboring unit from the top three units that optimize the total compactness of regions. Without considering compactness, we randomly add adjacent units to form a new region. The results show that the compactness-concerned region growing algorithm always identifies a larger number of regions for all thresholds (Figure 6). For example, the number of regions identified by the algorithm with compactness concern in the case of $T = 5,000$ ranges from 2,049 to 2,067 over 1,000 runs, whereas that by the algorithm without compactness concern varies from 1,980 to 2,001. In other words, accounting for compactness in the region growing phase

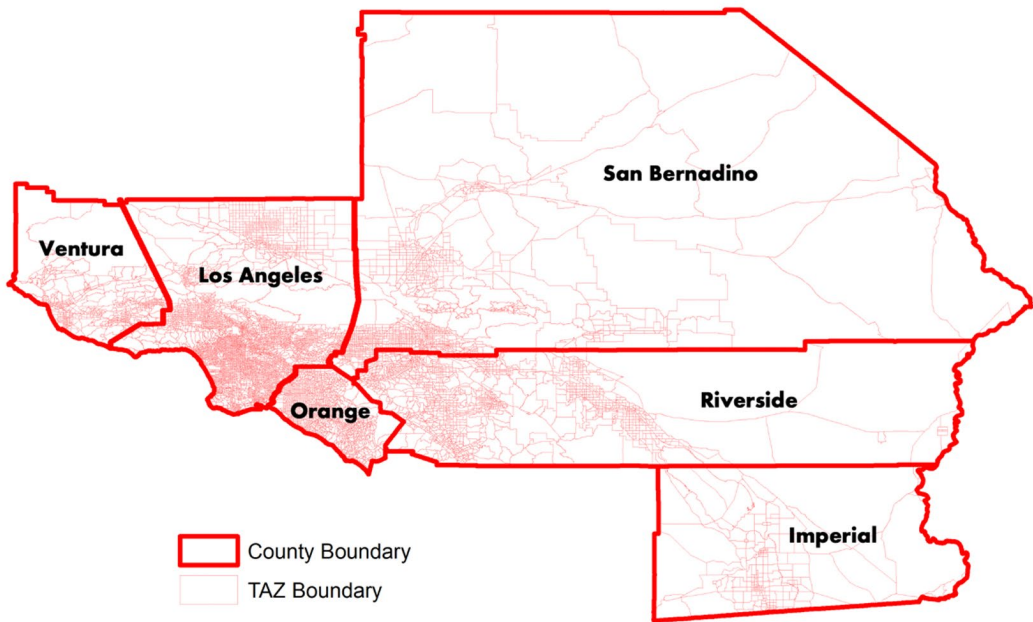


FIGURE 4 A study area of 111,670 TAZs in six southern California counties

can lead to a larger number of identified regions. Moreover, the average number of regions identified increases approximately 3.47, 16.6 and 17.9% when $T = 5,000$, 50,000, and 100,000 respectively, with compactness concern. That is, when the threshold is higher, compactness integration has a greater impact on identifying more regions.

After the phase of region growing and enclave assignment, we have a feasible partition for the study area, based on which a local search phase further improves the overall compactness of regions by locally swapping the neighboring unit areas. The average improvement of compactness and the corresponding computation time over 100 runs before and after the phase of local search for each threshold are summarized in Figure 7. The lower the threshold, the more pronounced the improvement of compactness in the local search phase. On the one hand, the compactness value is relatively small with $T = 5,000$ after phases of region growing and enclave assignment, which means there might be more scope to improve in the phase of local search. On the other hand, a lower threshold is always associated with a smaller number of units in each region. Thus, the compactness is more likely to improve by swapping units one by one between these small regions. For cases $T = 50,000$ and 100,000, the local search phase takes a long time, while its improvement of the compactness is limited. In these cases, the mean compactness of regions for a feasible partition is already higher than .8 after the region growing and enclave assignment phases. Given the quality of initial feasible solutions, the computation of further improvement via local search might not be justified due to its high computation cost.

6 | DISCUSSION

There are seven parameters in the solution approach. They are: the iterations of construction (IC) and the top N neighboring units (TNNU) in the region grow phase, the top N neighboring regions (TNNR) in the enclave assignment phase, and the iteration of simulated annealing (ISA), the temperature cooling rate α , the length of tabu list (LTL), and the maximum non-improving moves (MNIM) in the local search phase. To improve the

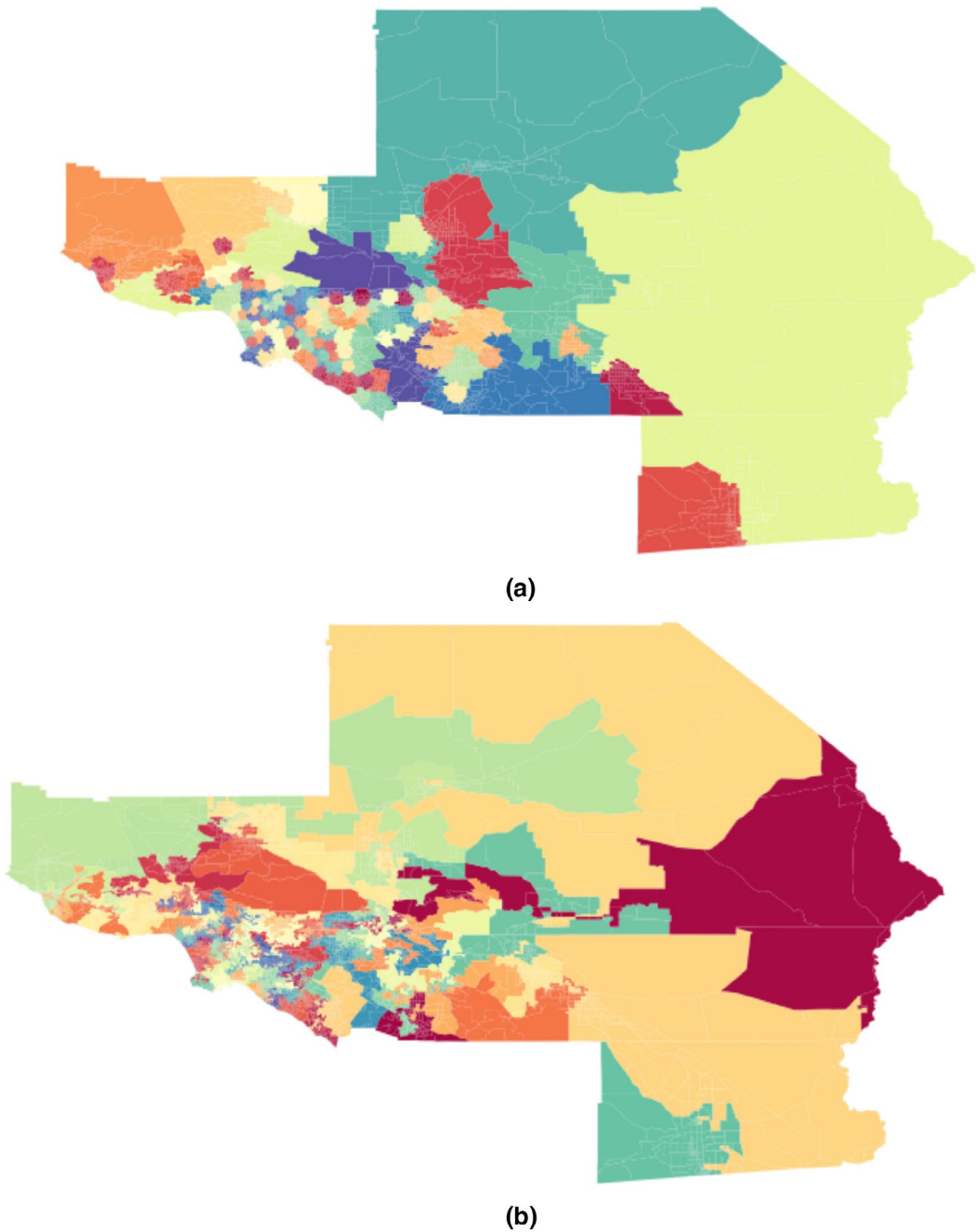


FIGURE 5 The best region partition plan obtained by solving: (a) max- p -compact-regions; and (b) max- p -regions

performance of the proposed solution, parameter selection can be derived by using some open-source Python packages. Parasweep is one option, which specifies each combination of parameters and then dispatches a parallel job for each parameter set to run the existing computational models. Another option is Hyperopt, which is more powerful for dealing with models with hyperparameters while you cannot specify the setting of parameters. The overall idea is to automatically search for the ideal combination of parameters based on a

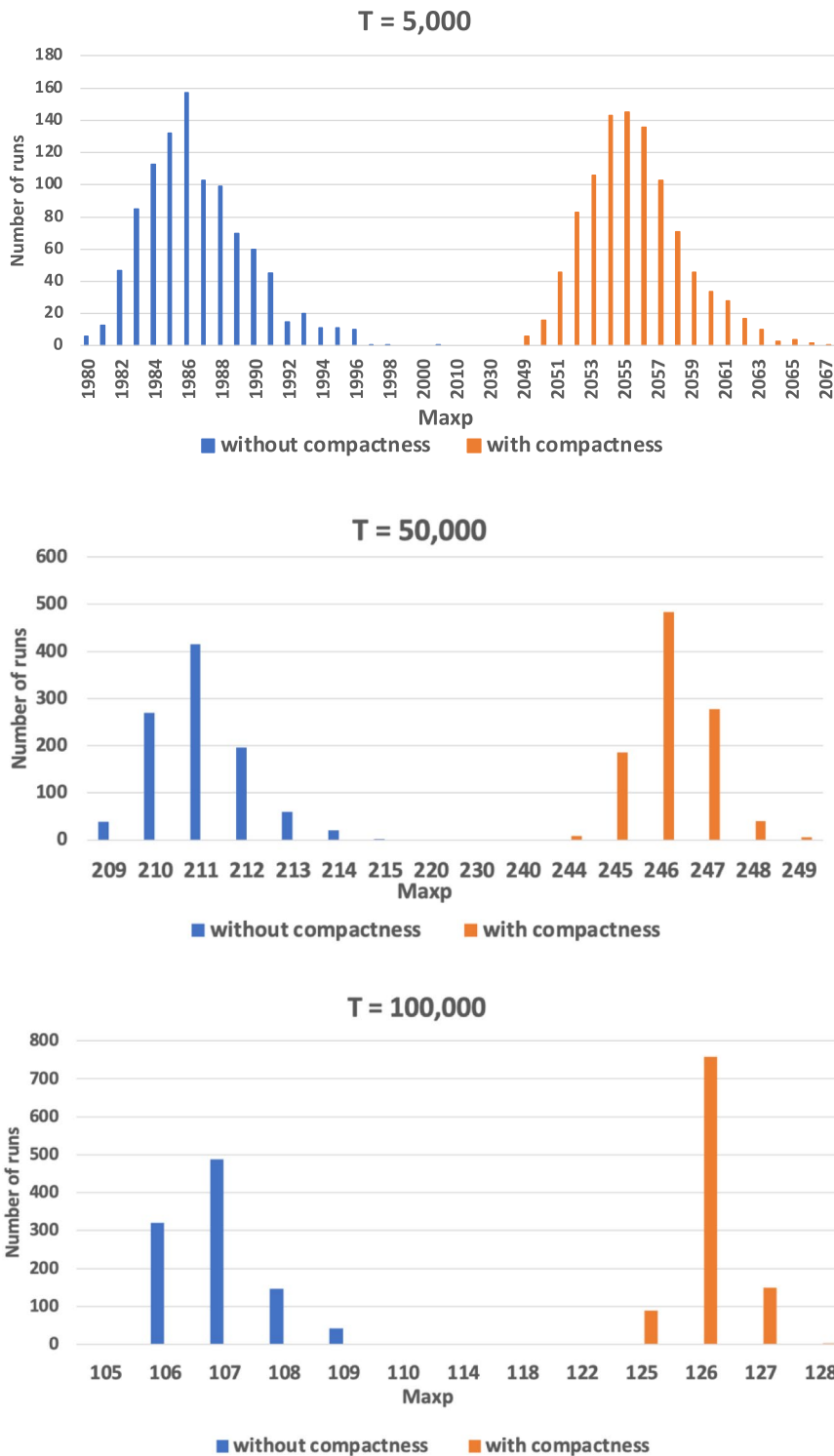


FIGURE 6 Distribution of the number of regions identified by the region growing algorithms with and without compactness

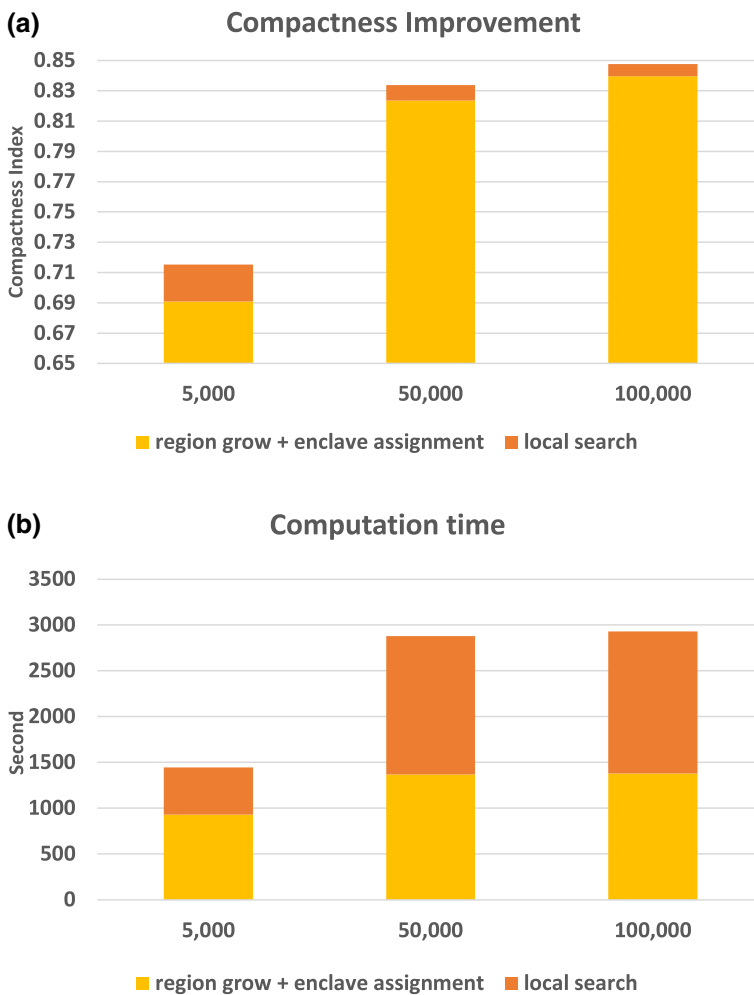


FIGURE 7 Distribution of: (a) the mean compactness index; and (b) computation time over 100 runs when $T = 5,000, 50,000$, and $100,000$

distributed Bayesian hyperparameter optimization method. Whatever option is chosen, the goal is to spend a little more time selecting the parameters in order to ensure the quality of solutions. The combinations of parameters selected by Parasweep were used in Section 5. Take the realistic data with $T = 100,000$ as an example; the parameter settings are $IC = 1,000$, $TNNU = 3$, $TNNR = 2$, $ISA = 100$, $\alpha = 0.998$, $LTL = 10$, and $MNM = 100$.

The definitions of these parameters may affect the performance of the solution approach in different data sets to varying degrees. Take $TNNU$ (in the region growth phase) as an example. When $N = 1$, the algorithm will only include the neighboring unit that maximizes compactness while gradually releasing the condition to allow more possibilities when N increases. Figure 8 records how the average compactness changes when $N = 1, \dots, 8$ in the region growing phase for different data sets and threshold combinations. In sum, the choice of N is more sensitive for the hypothetical data set, for which always adding the top candidate will optimize the quality of the solution. However, randomly picking one of the top three works best for the large realistic dataset.

Different from the original max- p -regions problem which accounts for the total within-region dissimilarity, the max- p -compact-regions problem considers the overall compactness of these regions (in Equation 10). However,

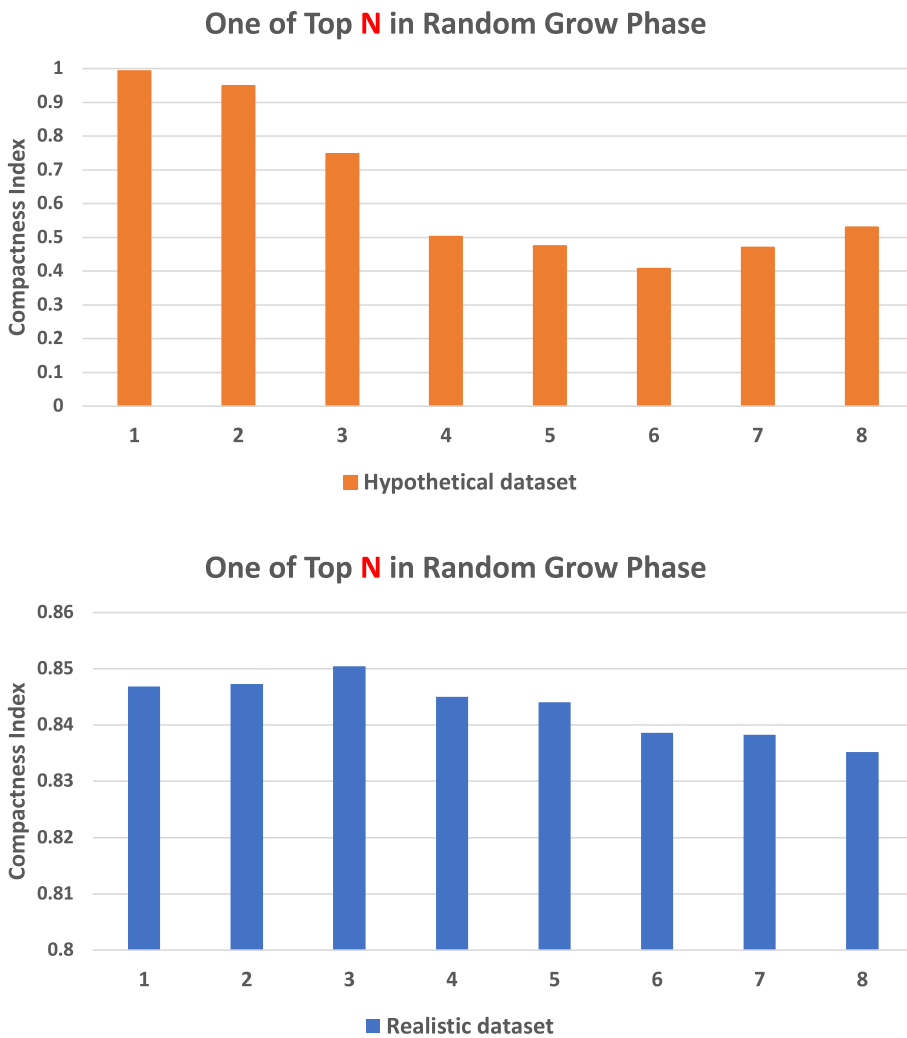


FIGURE 8 Changes of the solution when the TNNU parameter varies

the model and solution framework can be extended to a multiobjective regionalization problem that simultaneously deals with compactness and within-region dissimilarity. One potential new objective function is as follows:

$$\text{maximize } \sum_{k=1}^n \sum_{i=1}^n x_i^{k0} \times 10^{h''} + w_1 \times \sum_{k=1}^n C(Z_k) - w_2 \times \sum_i \sum_{j(j) \neq i} d_{ij} y_{ij} \quad (11)$$

where $w_1, w_2 \in [0, 1]$. They are weights linking the second term involving maximizing compactness and the third term associated with minimizing dissimilarity. Like h and h' , h'' is also a scaling factor, which is used to ensure the priority of finding a large number of identified regions before carrying out compactness and dissimilarity. An interesting extension for further research would be the exploration of this new multiobjective compact max- p model. The corresponding solution approach can follow the idea of our proposed one in Figure 1. The difference is that a certain form of integration of within-region dissimilarity and compactness needs to be calculated with each movement in the phase of enclave assignment and local search. The selection of w_1 and w_2 will affect the final partitioning solution, and it is challenging to quantify their relationship and explain the rationale behind it.

7 | CONCLUSIONS

In this article we have mathematically formulated a model, called the max- p -compact-regions problem, to consider compactness in regionalization when the number of regions, p , has not been predefined. The solution algorithm developed can efficiently relieve the computational intensity of the proposed problem. It is demonstrated to obtain global optimal solutions for a hypothetical max- p -compact-regions problem with a known optimal solution and to solve a large-scale practical regionalization problem. Results have been compared with perspectives of different data sets, different approaches to compactness in the solution process, and different parameter settings.

The critical findings show that considering compactness in the region growing phase can help to identify more regions and plays an essential role in improving the regions' compactness. The extent of this support may vary case by case in terms of threshold setting, attribute distribution, basic unit shape, and so on. This would be an interesting topic to explore in the future work. Moreover, for different data sets, the optimal settings of parameters may be different, and the smart selection of parameters will improve the performance of the solution method. The solution algorithm for the max- p -compact-regions problem will be included in the next release of *pysal/spopt* (<https://pysal.org/spopt/index.html>), an open-source Python library for solving optimization problems about regionalization, facility location, and transportation-oriented solutions with spatial data.

ACKNOWLEDGEMENTS

The authors thank colleagues at the Center for Geospatial Sciences for their support of this work.

CONFLICT OF INTEREST

No conflicts of interest are declared.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in <https://github.com/xf37/Max-p-compact-region>.

REFERENCES

- Assunção, R. M., Neves, M. C., Câmara, G., & da Costa Freitas, C. (2006). Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees. *International Journal of Geographical Information Science*, 20(7), 797–811. <https://doi.org/10.1080/13658810600665111>
- Camacho-Collados, M., Liberatore, F., & Angulo, J. M. (2015). A multi-criteria police districting problem for the efficient and effective design of patrol sector. *European Journal of Operational Research*, 246(2), 674–684. <https://doi.org/10.1016/j.ejor.2015.05.023>
- Datta, D., Malczewski, J., & Figueira, J. R. (2012). Spatial aggregation and compactness of census areas with a multiobjective genetic algorithm: A case study in Canada. *Environment and Planning B: Planning and Design*, 39(2), 376–392. <https://doi.org/10.1068/b38078>
- Duque, J. C., Anselin, L., & Rey, S. J. (2012). The max- p -regions problem. *Journal of Regional Science*, 52(3), 397–419. <https://doi.org/10.1111/j.1467-9787.2011.00743.x>
- Duque, J. C., Church, R. L., & Middleton, R. S. (2011). The p -regions problem. *Geographical Analysis*, 43(1), 104–126. <https://doi.org/10.1111/j.1538-4632.2010.00810.x>
- Duque, J. C., Ramos, R., & Suriñach, J. (2007). Supervised regionalization methods: A survey. *International Regional Science Review*, 30(3), 195–220. <https://doi.org/10.1177/0160017607301605>
- Folch, D. C., & Spielman, S. E. (2014). Identifying regions based on flexible user-defined constraints. *International Journal of Geographical Information Science*, 28(1), 164–184. <https://doi.org/10.1080/13658816.2013.848986>
- Fryer, R. G., Jr., & Holden, R. (2011). Measuring the compactness of political districting plans. *Journal of Law and Economics*, 54(3), 493–535. <https://doi.org/10.1086/661511>
- Ghiggi, C., Puliafito, P. P., & Zoppoli, R. (1975). A combinatorial method for healthcare districting. In J. Cea (Ed.), *Optimization techniques modeling and optimization in the service of man, Part 1: Proceedings* (Lecture Notes in Computer Science, Vol. 40, pp. 116–130). Berlin, Germany: Springer.

- Govorov, M., Becony  , G., Gienko, G., & Putrenko, V. (2019). Spatially constrained regionalization with multilayer perceptron. *Transactions in GIS*, 23(5), 1048–1077. <https://doi.org/10.1111/tgis.12557>
- Guo, D. (2008). Regionalization with dynamically constrained agglomerative clustering and partitioning (REDCAP). *International Journal of Geographical Information Science*, 22(7), 801–823. <https://doi.org/10.1080/13658810701674970>
- Guo, D., & Wang, H. (2011). Automatic region building for spatial analysis. *Transactions in GIS*, 15, 29–45. <https://doi.org/10.1111/j.1467-9671.2011.01269.x>
- Hamidi, S., & Zandiatashbar, A. (2021). Compact development and adherence to stay-at-home order during the COVID-19 pandemic: A longitudinal investigation in the United States. *Landscape and Urban Planning*, 205, 103952. <https://doi.org/10.1016/j.landurbplan.2020.103952>
- Hess, S. W., & Samuels, S. A. (1971). Experiences with a sales districting model: Criteria and implementation. *Management Science*, 18(4-Part 2), P-41–P-54. <https://doi.org/10.1287/mnsc.18.4.P41>
- Hess, S. W., Weaver, J. B., Siegfeldt, H. J., Whelan, J. N., & Zitlau, P. A. (1965). Nonpartisan political redistricting by computer. *Operations Research*, 13(6), 998–1006. <https://doi.org/10.1287/opre.13.6.998>
- Li, W., Church, R. L., & Goodchild, M. F. (2014). The p -compact-regions problem. *Geographical Analysis*, 46(3), 250–273. <https://doi.org/10.1111/gean.12038>
- Li, W., Goodchild, M. F., & Church, R. (2013). An efficient measure of compactness for two-dimensional shapes and its application in regionalization problems. *International Journal of Geographical Information Science*, 27(6), 1227–1250. <https://doi.org/10.1080/13658816.2012.752093>
- Li, Y., Shi, Z., Li, F., & Li, H. Y. (2007). Delineation of site-specific management zones using fuzzy clustering analysis in a coastal saline land. *Computers and Electronics in Agriculture*, 56(2), 174–186. <https://doi.org/10.1016/j.compag.2007.01.013>
- Niemi, R. G., Grofman, B., Carlucci, C., & Hofeller, T. (1990). Measuring compactness and the role of a compactness standard in a test for partisan and racial gerrymandering. *Journal of Politics*, 52(4), 1155–1181. <https://doi.org/10.2307/2131686>
- Niesterowicz, J., Stepinski, T. F., & Jasiewicz, J. (2016). Unsupervised regionalization of the United States into landscape pattern types. *International Journal of Geographical Information Science*, 30(7), 1450–1468. <https://doi.org/10.1080/13658816.2015.1134796>
- Oehrl  n, J., & Haunert, J. H. (2017). A cutting-plane method for contiguity-constrained spatial aggregation. *Journal of Spatial Information Science*, 15, 89–120. <https://doi.org/10.5311/JOSIS.2017.15.379>
- Rey, S. J., Anselin, L., Folch, D. C., Arribas-Bel, D., Sastre Guti  rrez, M. L., & Interlante, L. (2011). Measuring spatial dynamics in metropolitan areas. *Economic Development Quarterly*, 25(1), 54–64. <https://doi.org/10.1177/0891242410383414>
- She, B., Duque, J. C., & Ye, X. (2017). The network-max- p -regions model. *International Journal of Geographical Information Science*, 31(5), 962–981. <https://doi.org/10.1080/13658816.2016.1252987>
- Spielman, S. E., & Singleton, A. (2015). Studying neighborhoods using uncertain data from the American community survey: A contextual approach. *Annals of the Association of American Geographers*, 105(5), 1003–1025. <https://doi.org/10.1080/00045608.2015.1052335>
- Weaver, J. B., & Hess, S. W. (1963). A procedure for nonpartisan districting: Development of computer techniques. *Yale Law Journal*, 73, 288. <https://doi.org/10.2307/794769>
- Wei, R., Rey, S., & Knaap, E. (2021). Efficient regionalization for spatially explicit neighborhood delineation. *International Journal of Geographical Information Science*, 35(1), 135–151. <https://doi.org/10.1080/13658816.2020.1759806>

How to cite this article: Feng, X., Rey, S., & Wei, R. (2022). The max- p -compact-regions problem. *Transactions in GIS*, 26, 717–734. <https://doi.org/10.1111/tgis.12874>