Constrained Clustering of Territories in the Context of Car Insurance

Samuel Perreault
Jean-Philippe Le Cavalier

Laval University

July 2014

Context : car insurance in Ontario

Important factor used : location of the residence

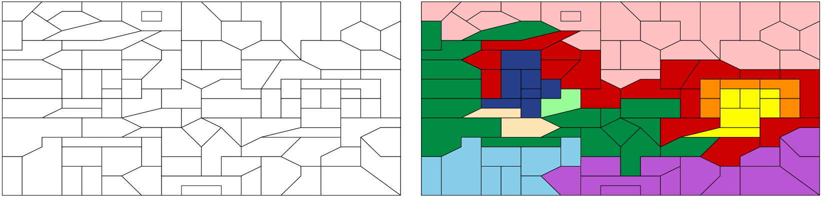
"One of the concerns from a public policy perspective is that if a territory is based on a small geographical area, even though densely populated, socio-economic factors may be influencing loss costs. In addition, drivers may operate their vehicles all over the city, so narrowly defined territories may not be logical." (FCSO Auto Bulletin No. A- 01/05)

New regulations imposed by The Financial Services Commission of Ontario

1. a territory must possess a minimal exposure of 7,500 cars over a three-year span ;
2. there must be no more than 55 territories in Ontario, of which no more than 10 should cover a fraction of the greater Toronto ;
3. all territories must be contiguous.

The challenge :

The challenge is to group the initial territories in such a way that the resulting territories are as homogeneous as possible.



The goal is to minimize intra-group variation : $D := \sum_i w_i(\lambda_i - y_i)^2$.

- w_i is the exposure associated to territory i ;
- λ_i is the initial premium associated to territory i ;
- y_i is the adjusted premium associated to territory i (final premium).

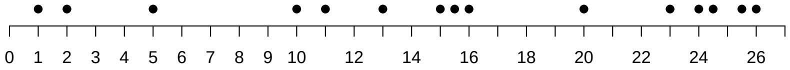
Section 2

A typical k-means algorithm

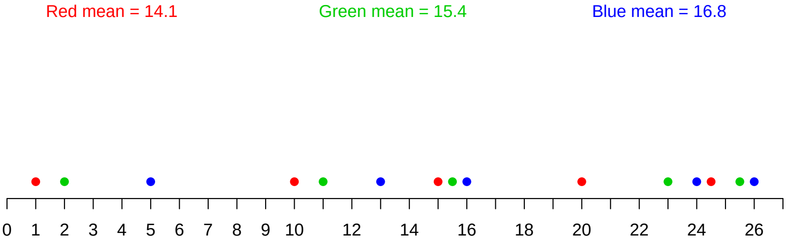
Iterative steps [Steinley, 2006] : K groups

1. Initialization : formation of the initial groups on which the algorithm will be applied over and over again. Many existing methods :
   - randomly select K points to be the initial centroids ;
   - randomly assign each point to one of the K groups (Forgy method) ;
   - use a hierarchical method to form K groups.
2. Group centroids (means) are obtained for each group.
3. Points are compared to each centroid and moved to the group whose centroid is closest (optimization).
4. Steps 2 and 3 are repeated until no points can be moved between groups.

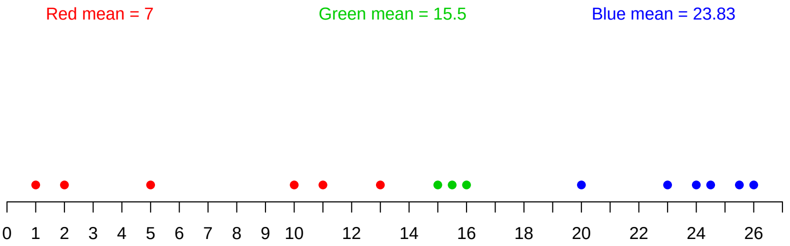
Basic example



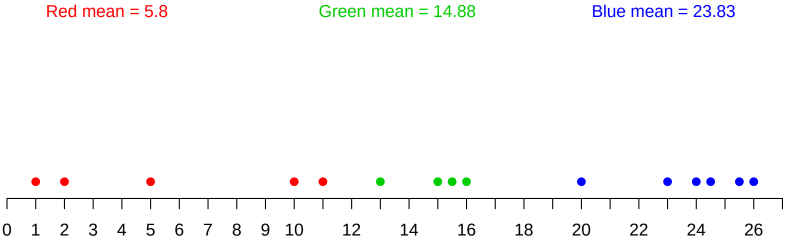
Basic example



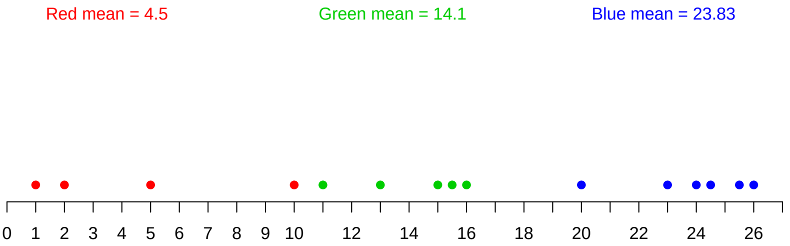
Basic example



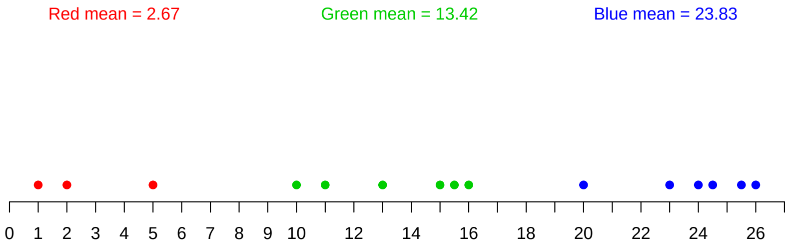
Basic example



Basic example



Basic example



Section 3

The modified k-means algorithm

Modification 1 : initialization

Problem

The contiguity constraint does not allow the use of any of the initialization methods presented.

Modification

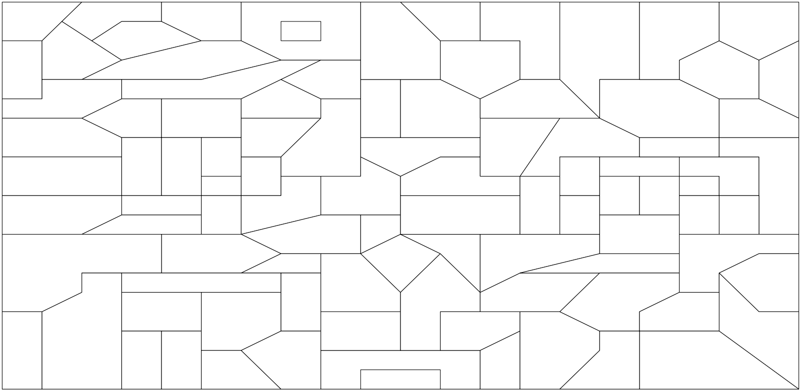
1. Randomly (or not) choose some territory to be the "center" (vs centroid) of a group.
2. Verify that the minimal exposure constraint is satisfied. If it is, go to step 3 ; if not, randomly choose a (unassigned) neighbor of this territory/group, merge it and repeat step 2.
3. Repeat the steps 1 and 2 until the K centers have been chosen.
4. Iteratively assign all territories adjacent to a group to it. If an unassigned territory is adjacent to more than one group, assign it randomly.

Example 1

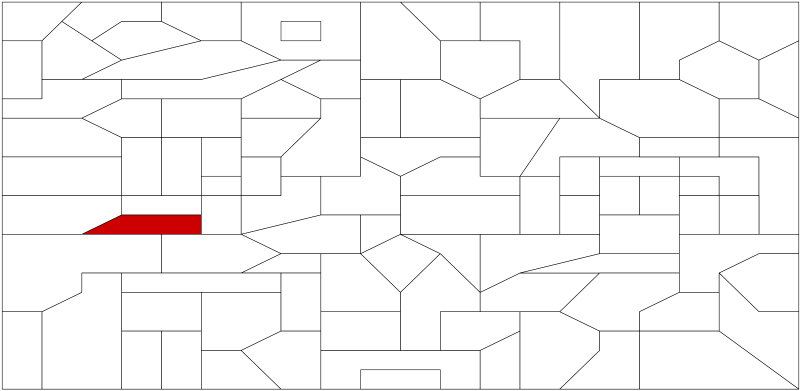
Initialization

(random selection)

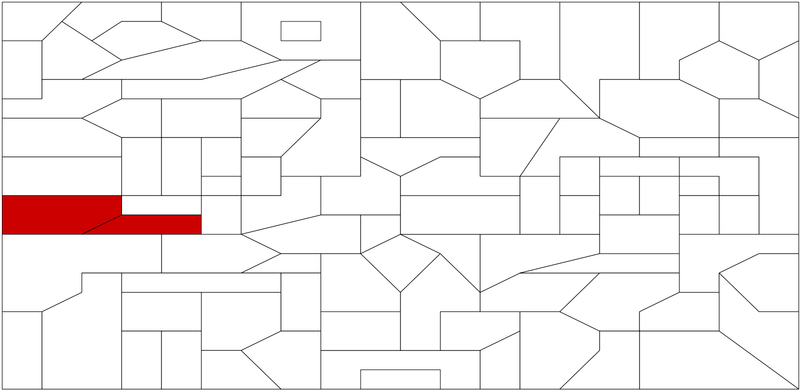
Example 1



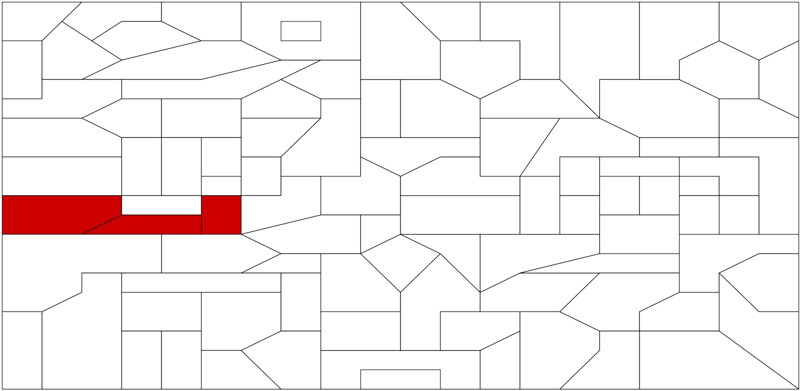
Example 1



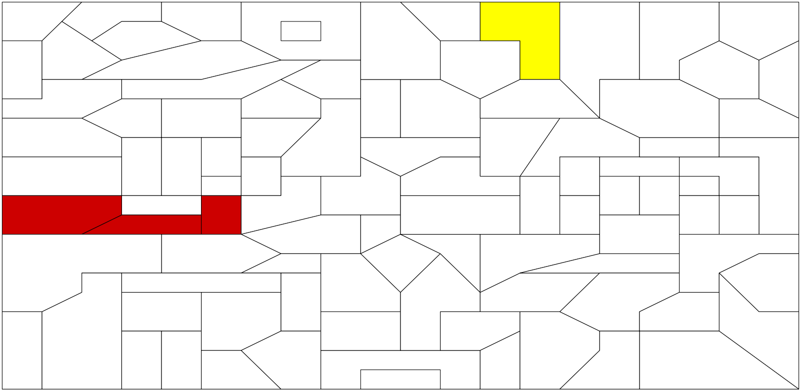
Example 1



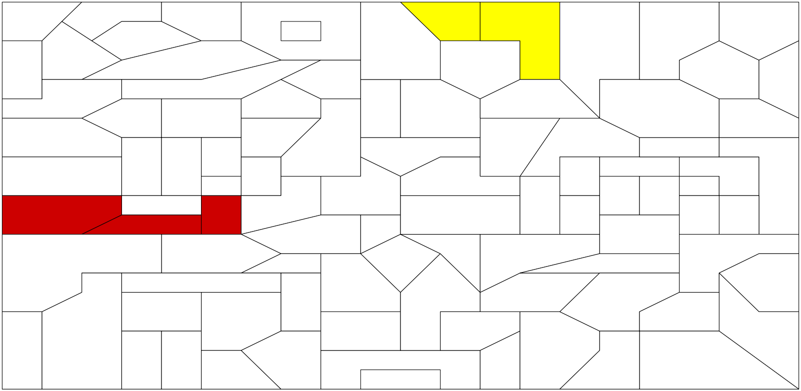
Example 1



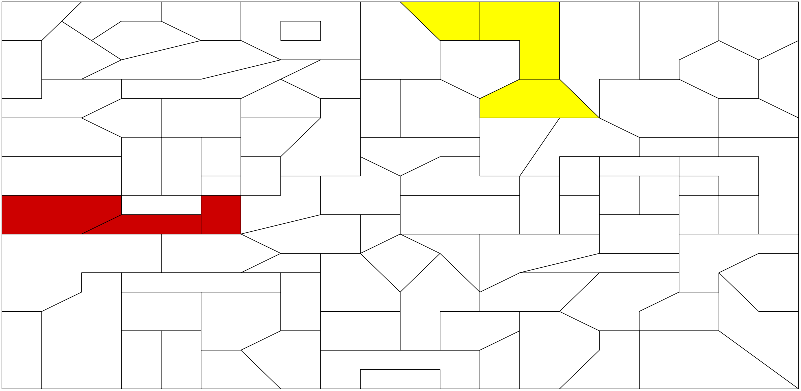
Example 1



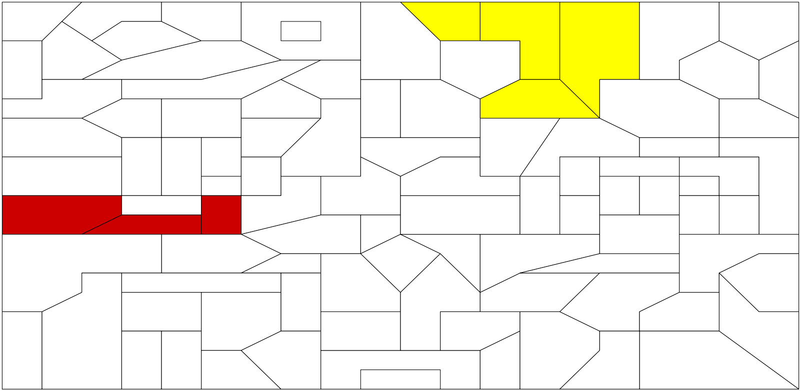
Example 1



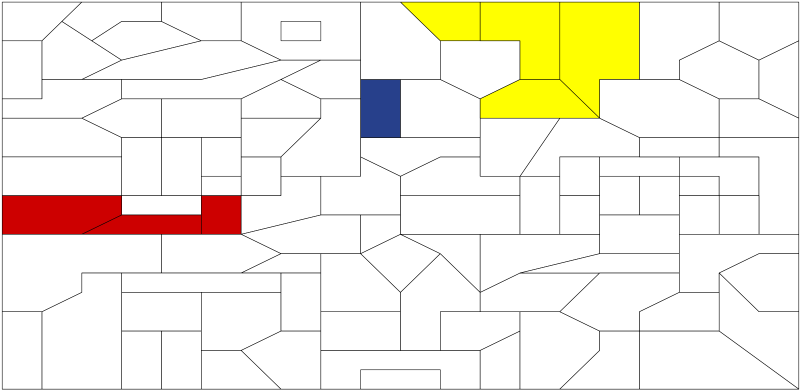
Example 1



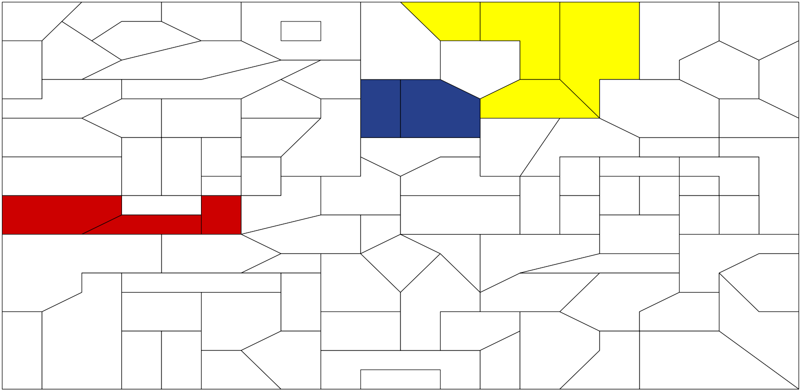
Example 1



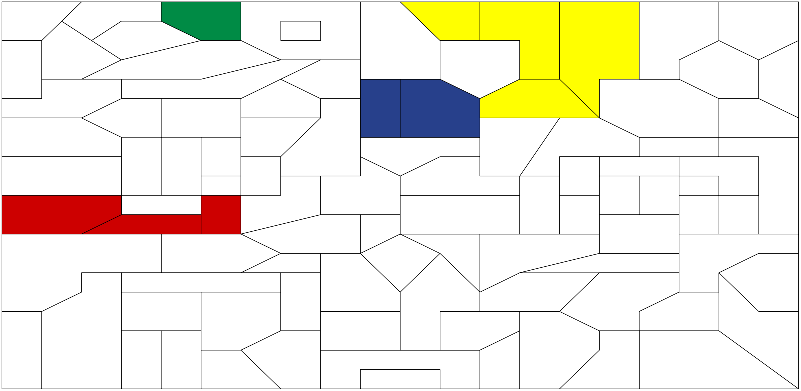
Example 1



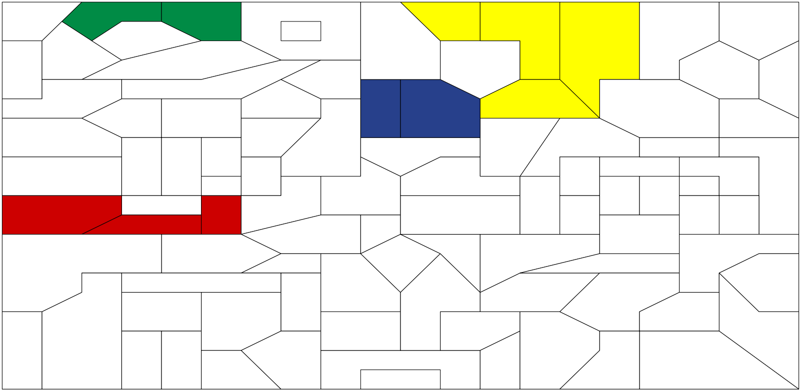
Example 1



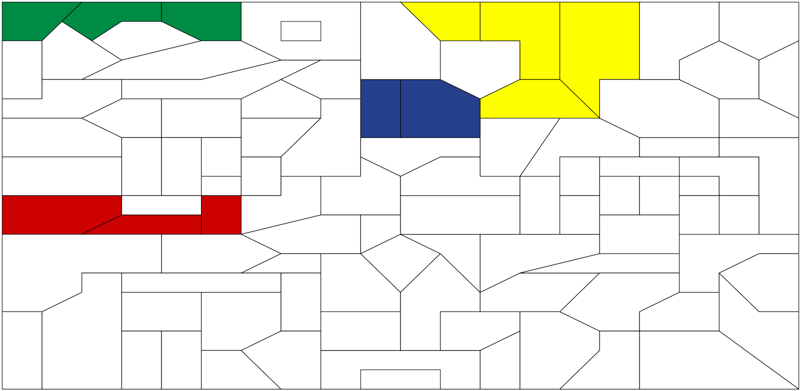
Example 1



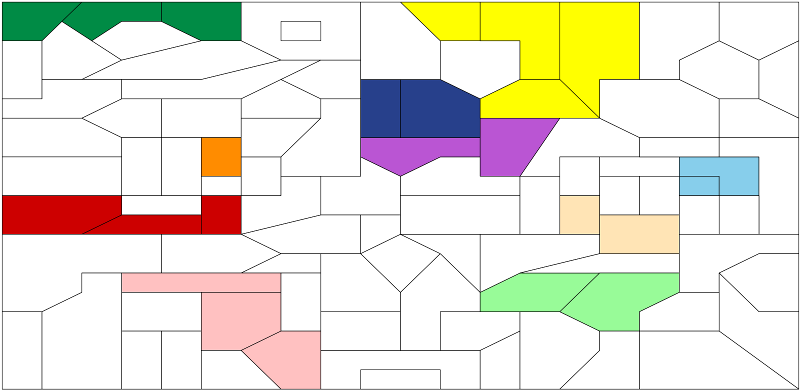
Example 1



Example 1



Example 1

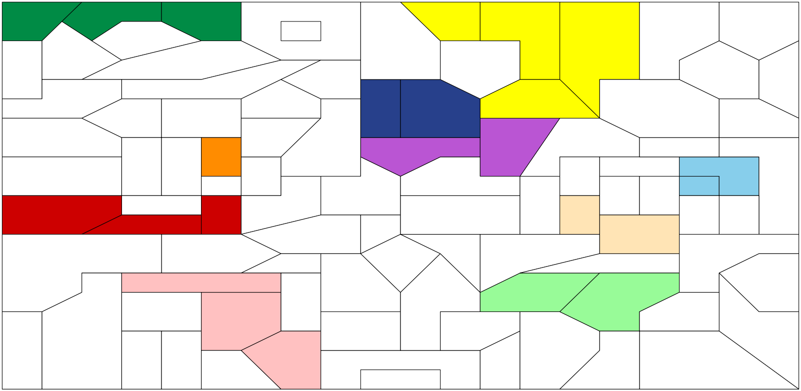


Example 1

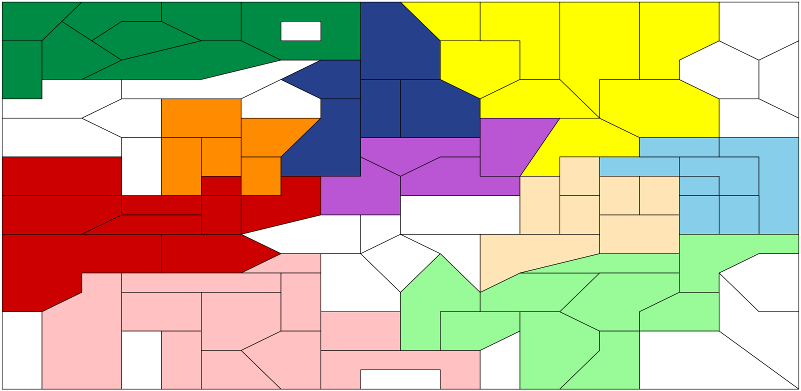
Initialization

(expansion)

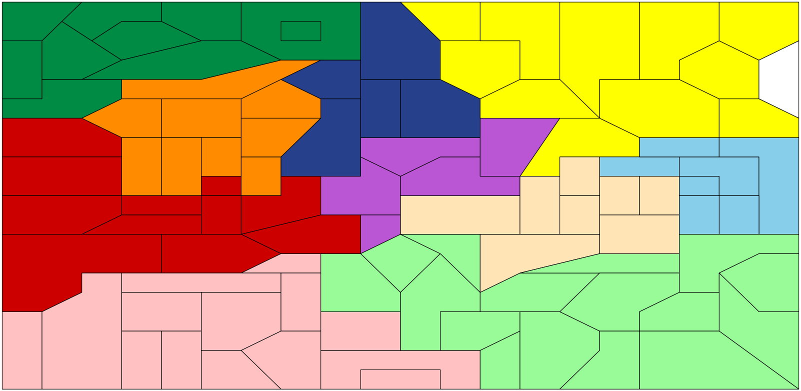
Example 1



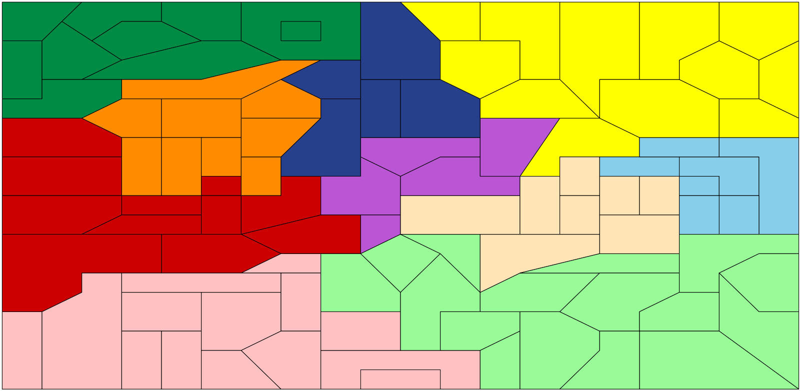
Example 1



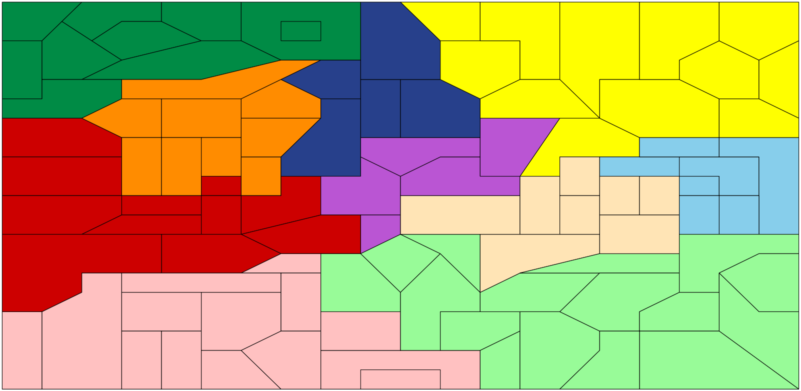
Example 1



Example 1

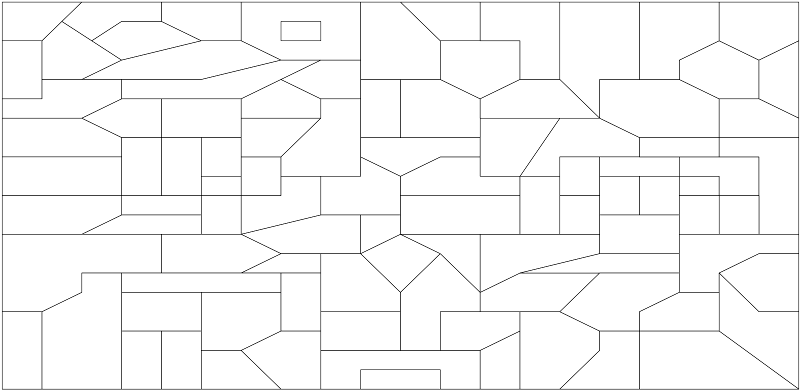


Example 1

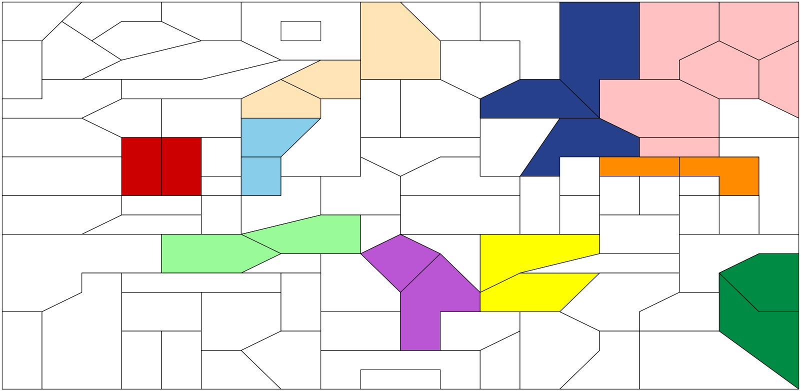


$$D = 510,771,107$$

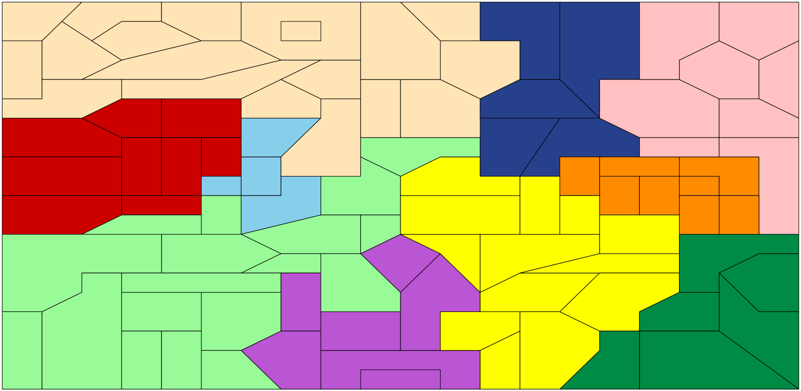
Example 2



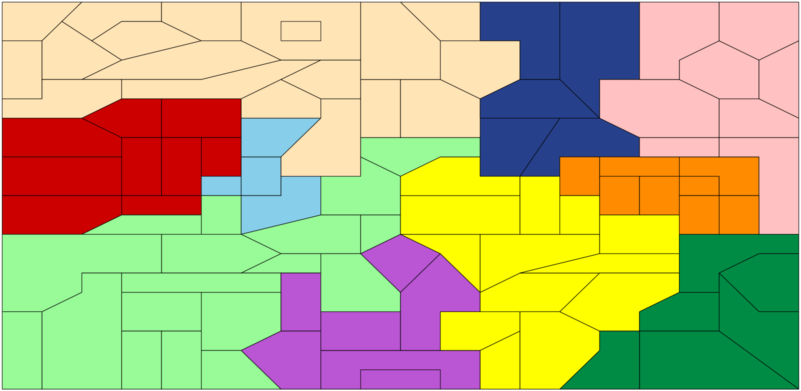
Example 2



Example 2



Example 2



$$D = 532,487,853$$

Modification 2 : reassignment - optimization

Problem

During the reassignment of the groups in the k-means algorithm, a territory is placed in the group with the closest centroid. In the present context, such a reassignment would often lead to the violation of the contiguity constraint.

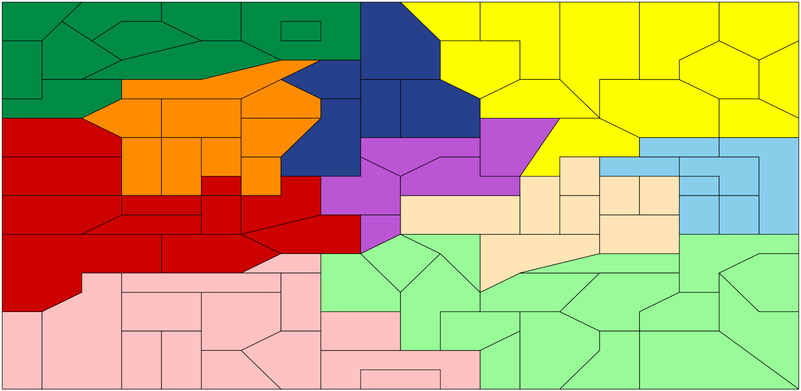
Modification

The only "moveable" territories are the ones on the borders. Moreover, the number of groups to which they can belong is restrained to groups adjacent to them and, of course, the group to which they belong.

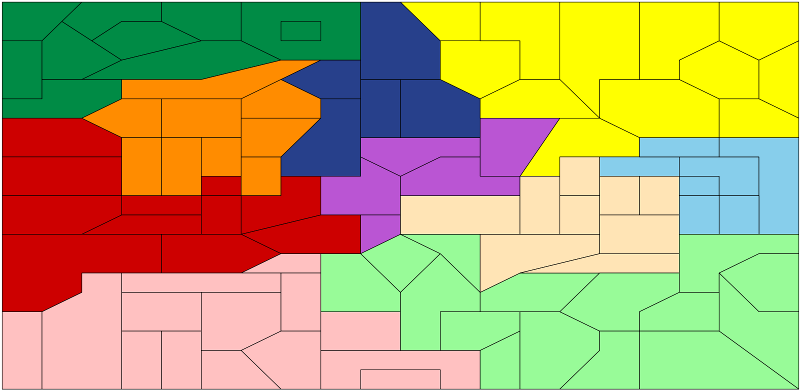
Example 1 (cont'd)

Optimization

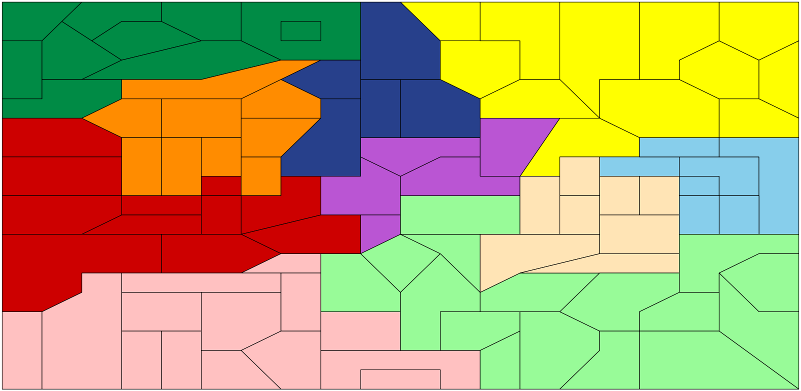
Example 1 (cont'd)



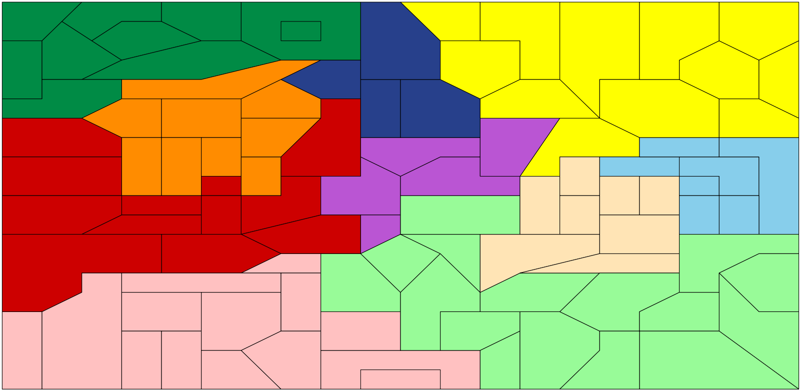
Example 1 (cont'd)



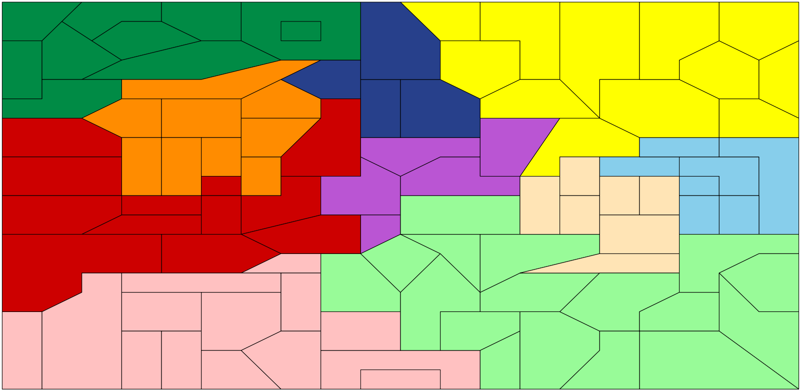
Example 1 (cont'd)



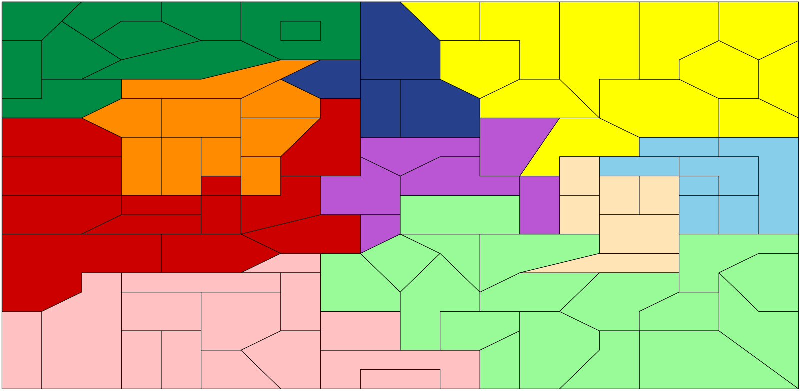
Example 1 (cont'd)



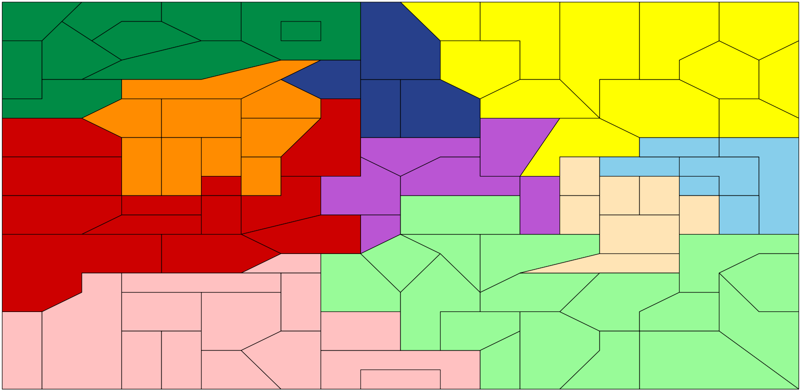
Example 1 (cont'd)



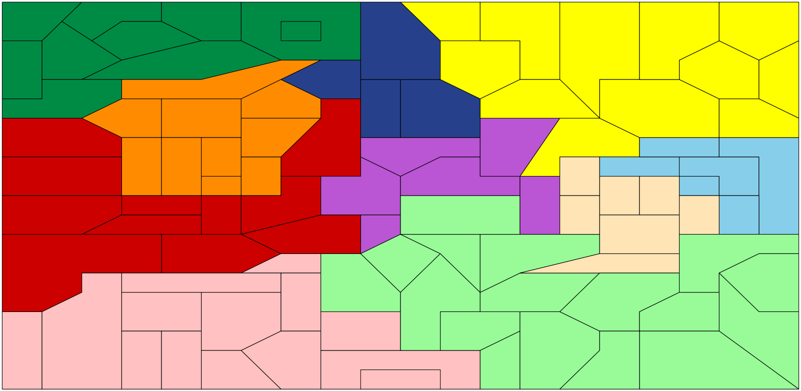
Example 1 (cont'd)



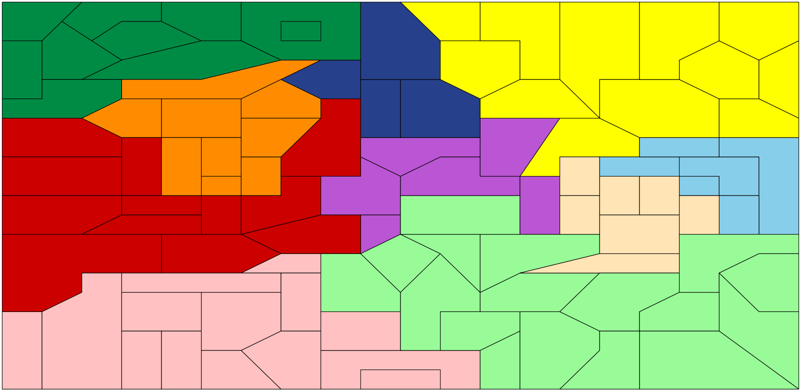
Example 1 (cont'd)



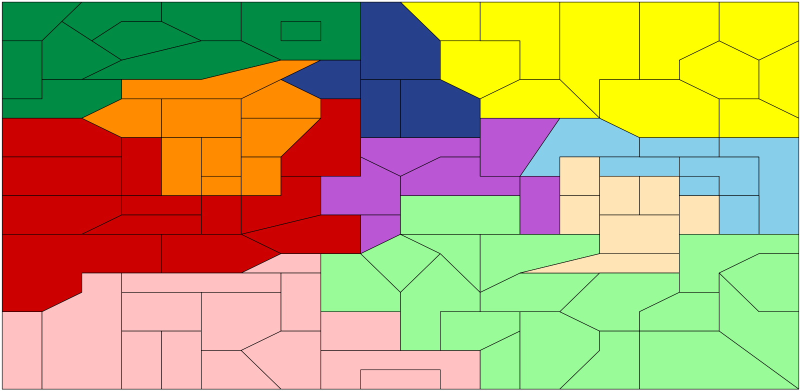
Example 1 (cont'd)



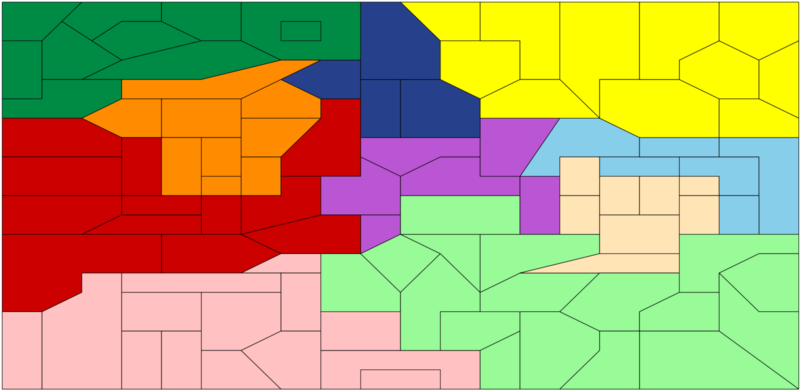
Example 1 (cont'd)



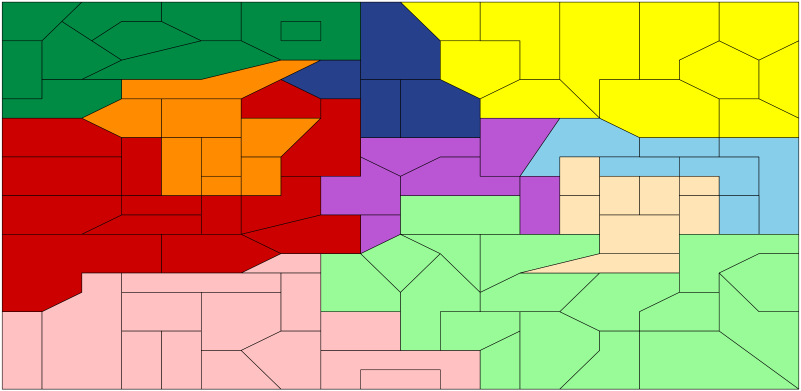
Example 1 (cont'd)



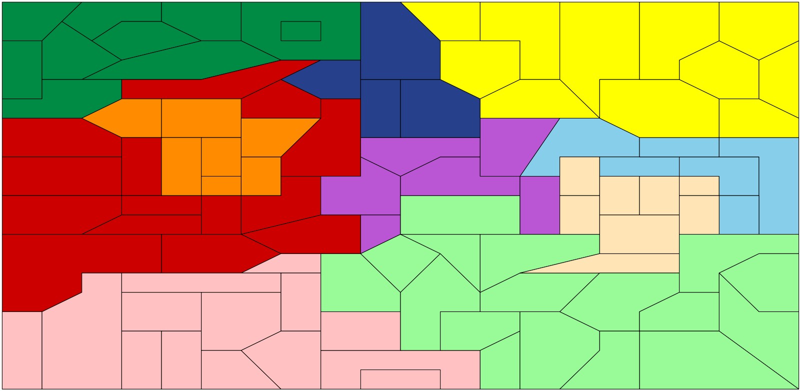
Example 1 (cont'd)



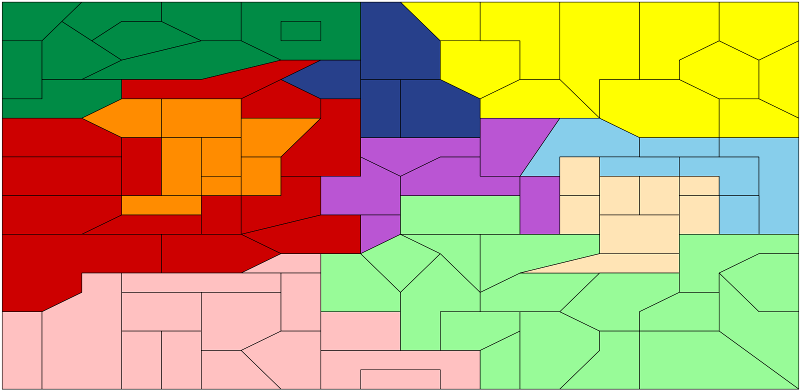
Example 1 (cont'd)



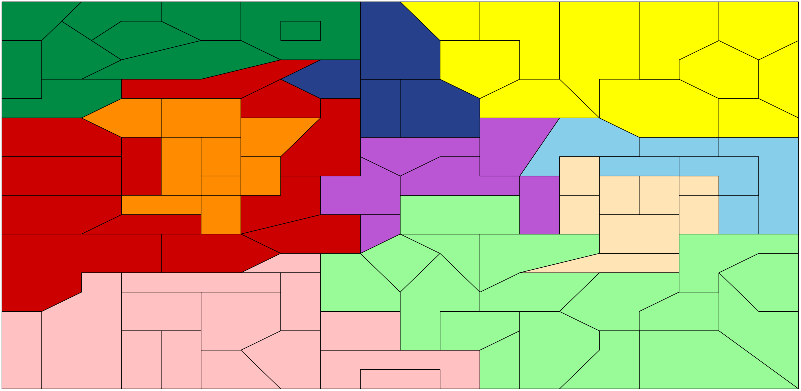
Example 1 (cont'd)



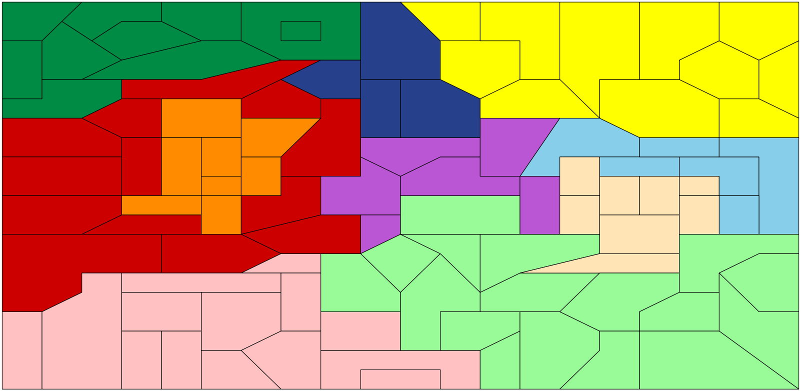
Example 1 (cont'd)



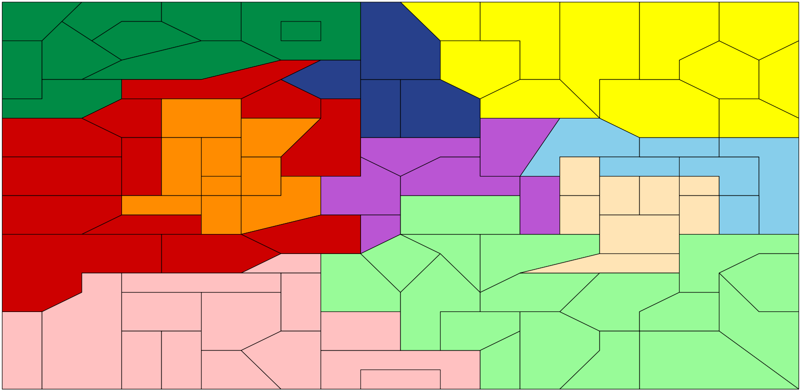
Example 1 (cont'd)



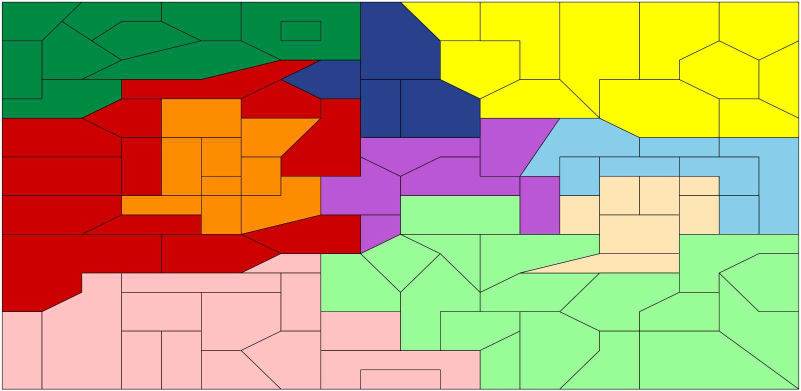
Example 1 (cont'd)



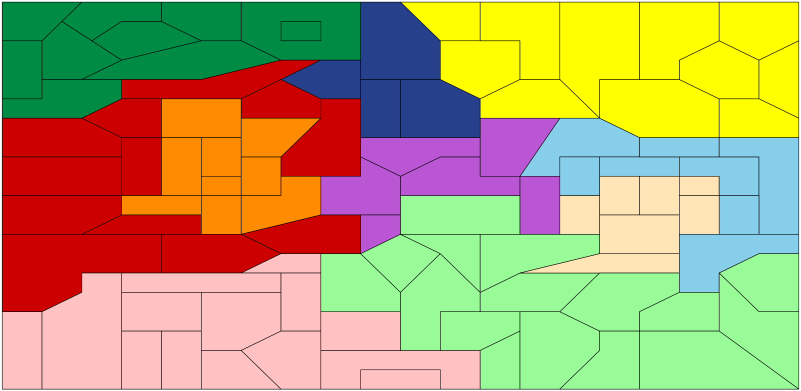
Example 1 (cont'd)



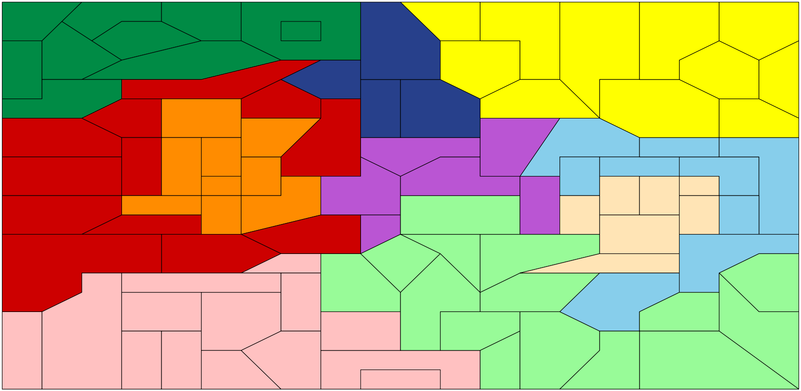
Example 1 (cont'd)



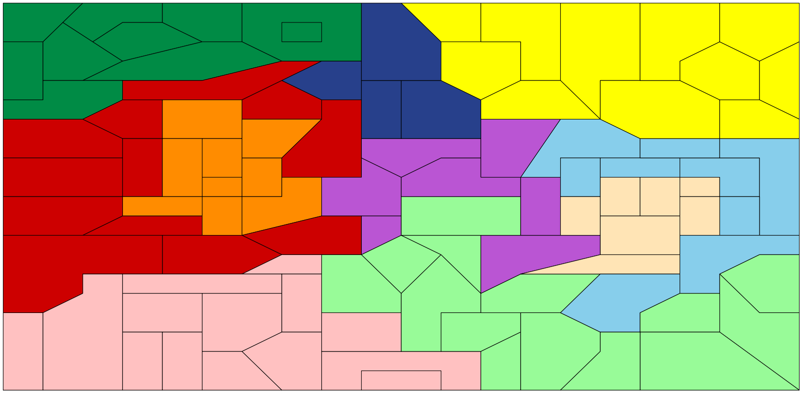
Example 1 (cont'd)



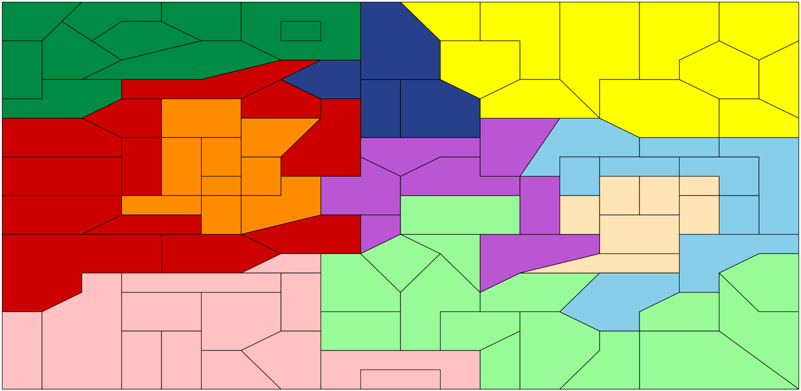
Example 1 (cont'd)



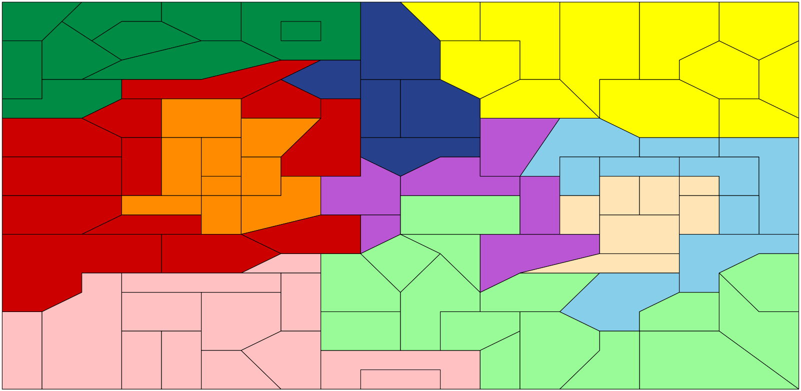
Example 1 (cont'd)



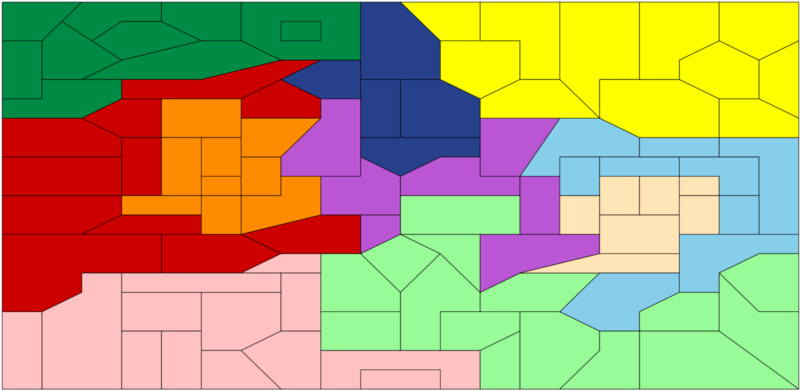
Example 1 (cont'd)



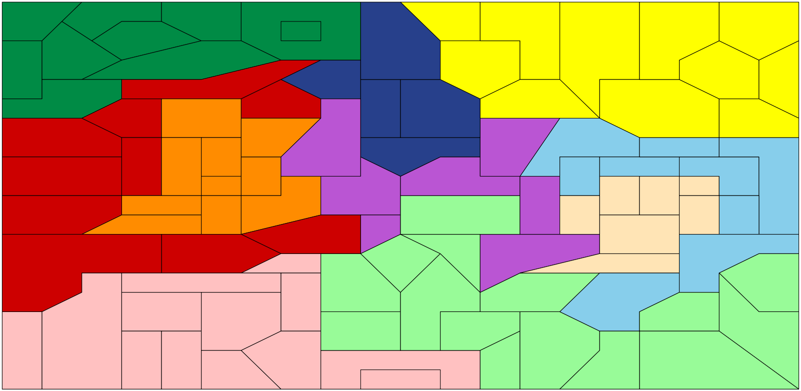
Example 1 (cont'd)



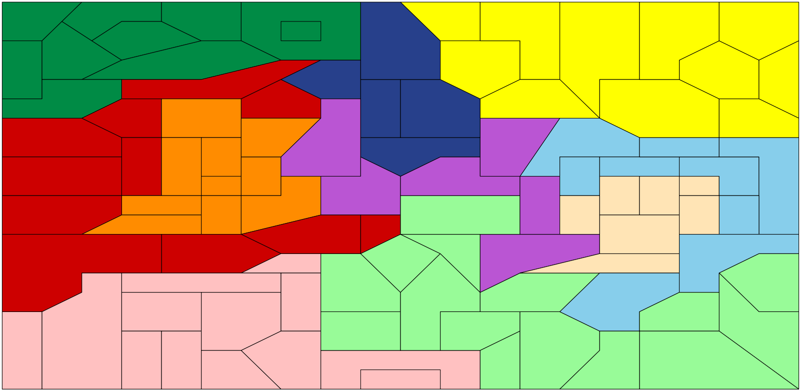
Example 1 (cont'd)



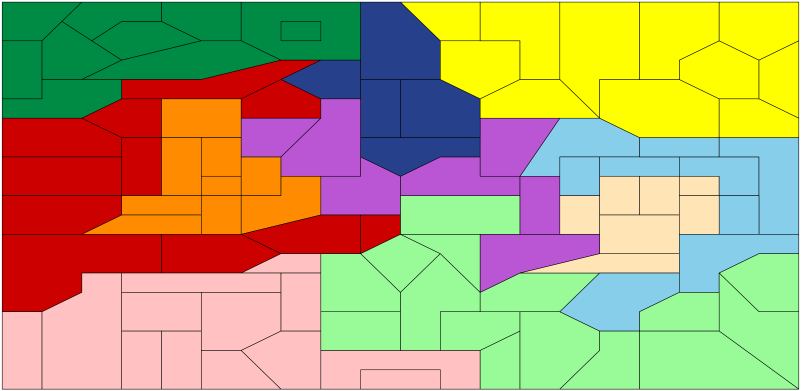
Example 1 (cont'd)



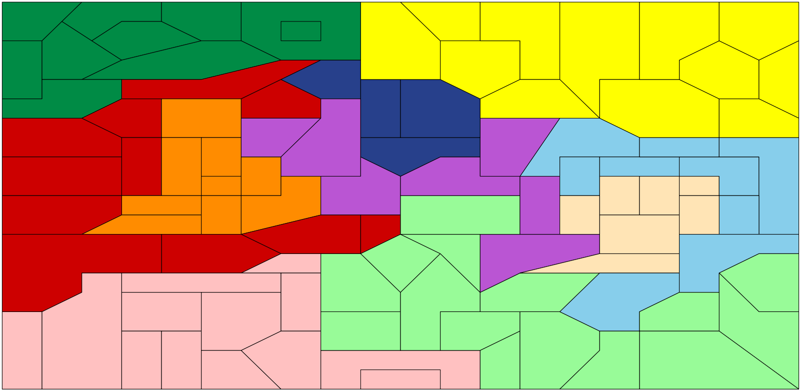
Example 1 (cont'd)



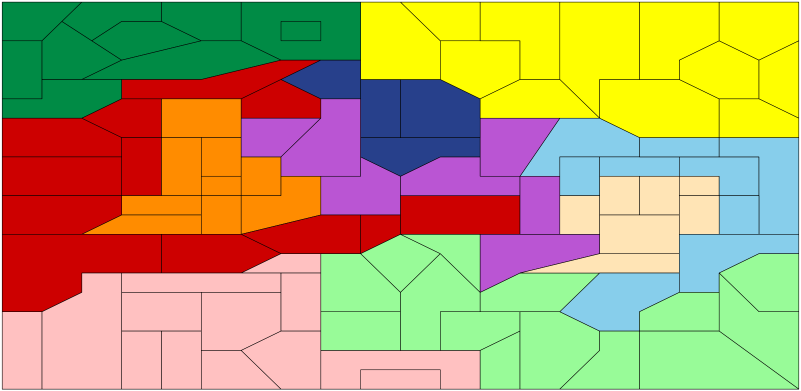
Example 1 (cont'd)



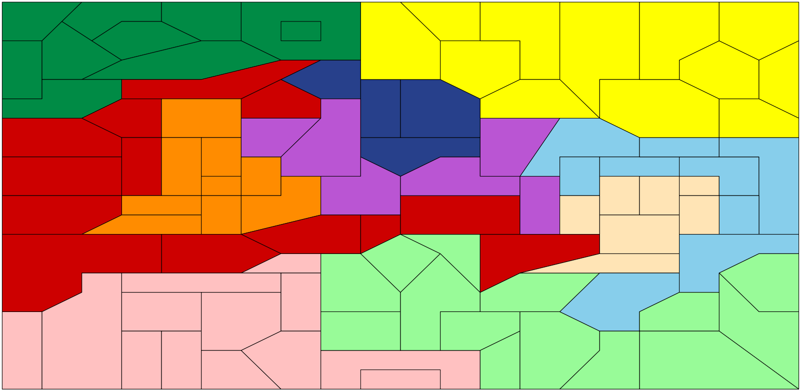
Example 1 (cont'd)



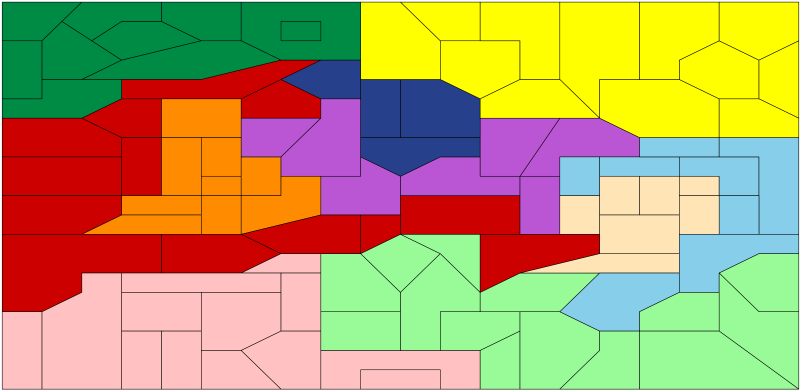
Example 1 (cont'd)



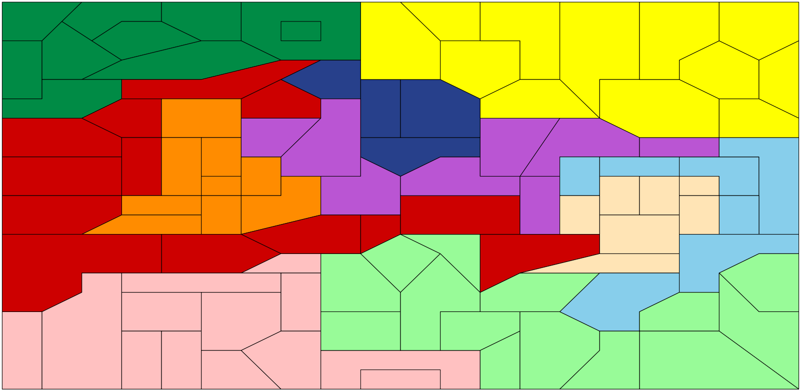
Example 1 (cont'd)



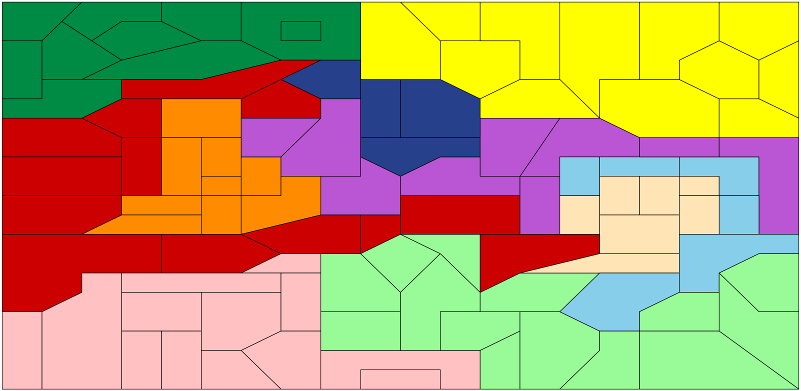
Example 1 (cont'd)



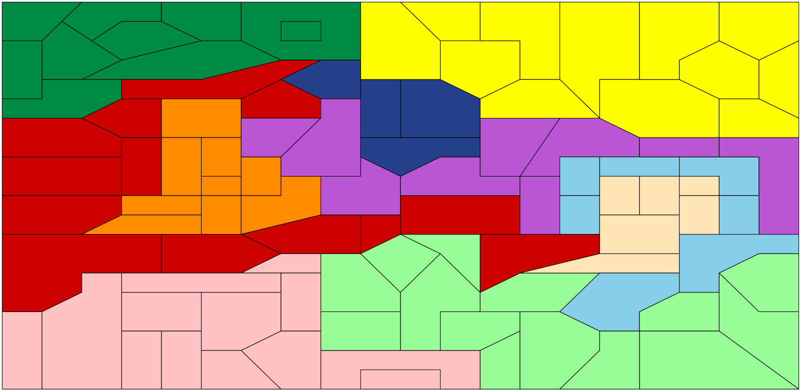
Example 1 (cont'd)



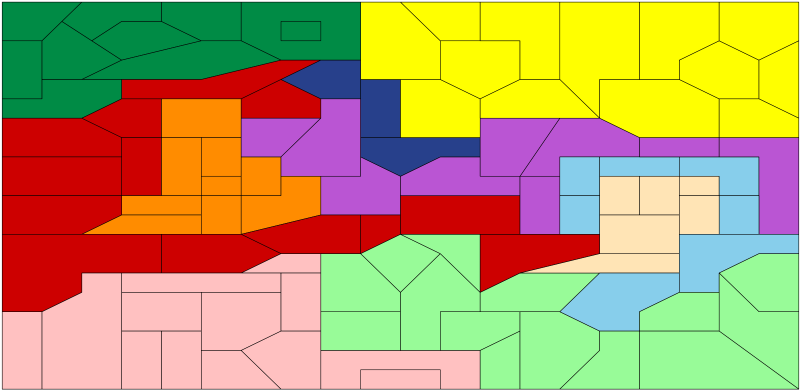
Example 1 (cont'd)



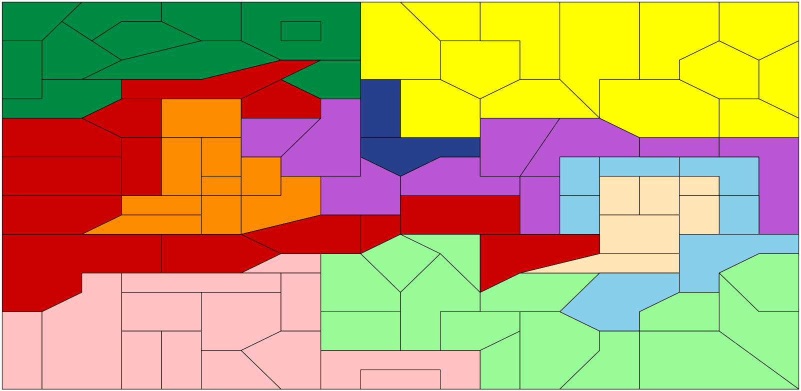
Example 1 (cont'd)



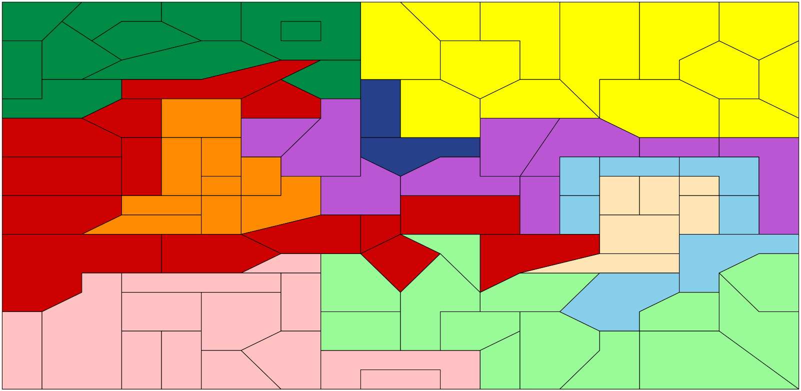
Example 1 (cont'd)



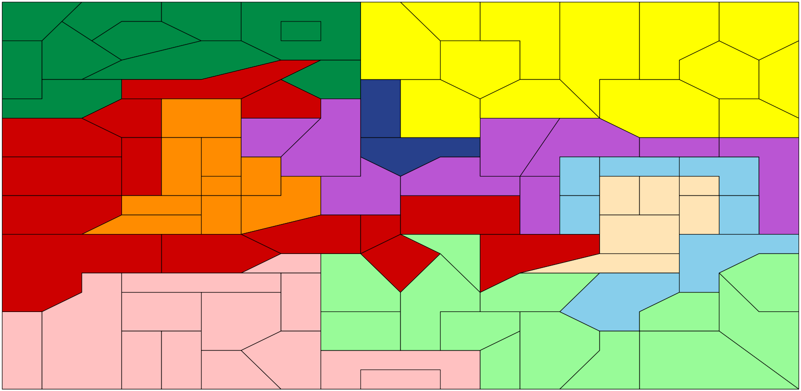
Example 1 (cont'd)



Example 1 (cont'd)

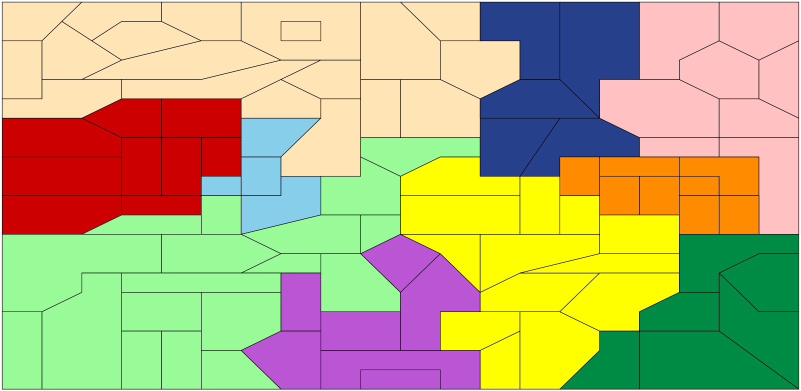


Example 1 (cont'd)

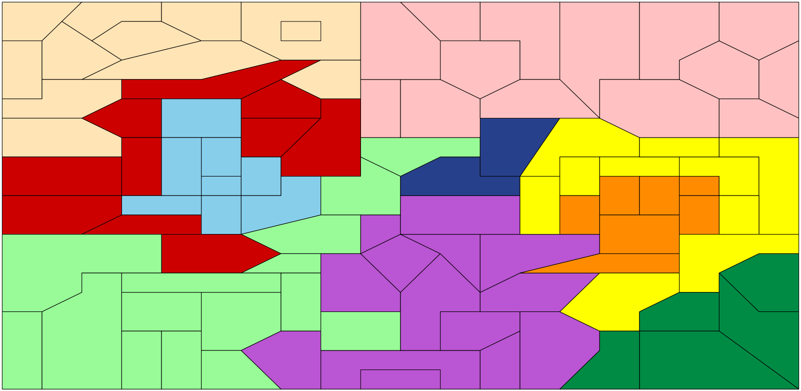


$$D = 69,066,748$$

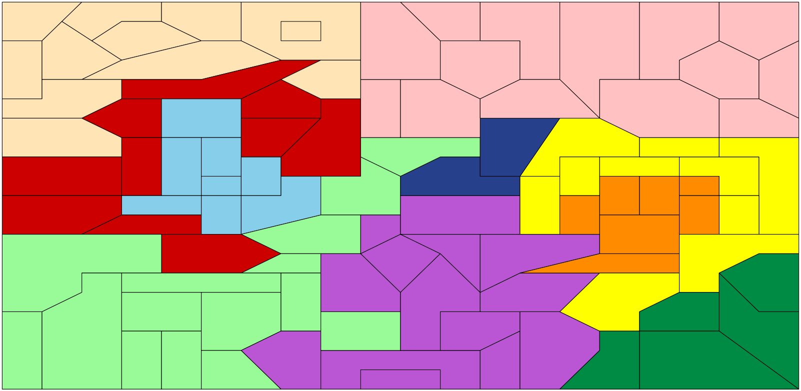
Example 2 (cont'd)



Example 2 (cont'd)



Example 2 (cont'd)



$$D = 108,263,053$$

Section 4

Some other considerations and an additional step

Complications

Complication due to the minimal exposure constraint

Sometimes, a group diminishes until it cannot "give" any more territories : switching a territory would lead to the violation of the minimal exposure constraint.

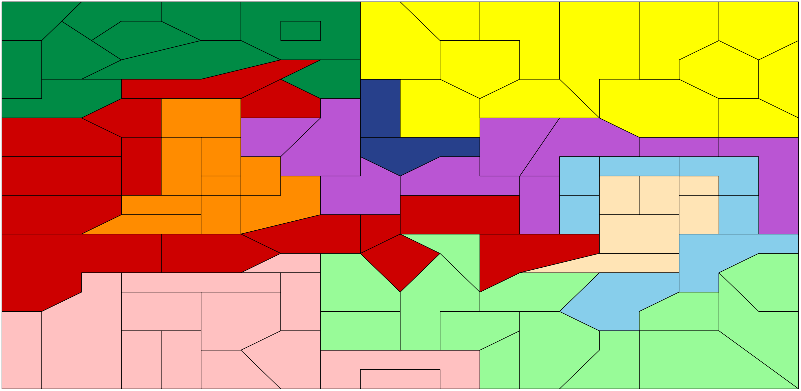
- It may be advantageous to merge this (potentially negligible) group to a neighbor and add a group somewhere else.

Similarity between adjacent groups

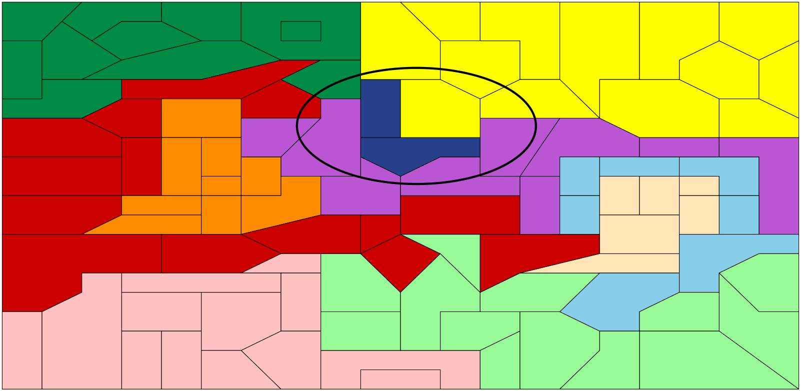
Sometimes, the map is stable and two adjacent groups are quite similar. It is possible that some other groups are very heterogeneous, i.e. formed by territories different from one another.

- In that case, it would be advantageous to merge the two similar groups and split the heterogeneous one.

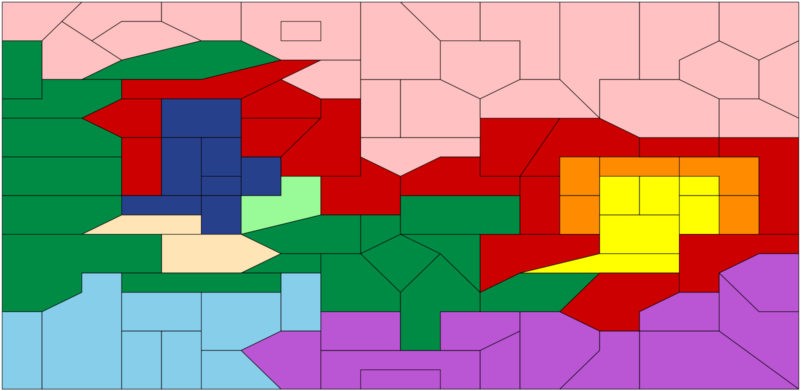
Complication due to the minimal exposure constraint



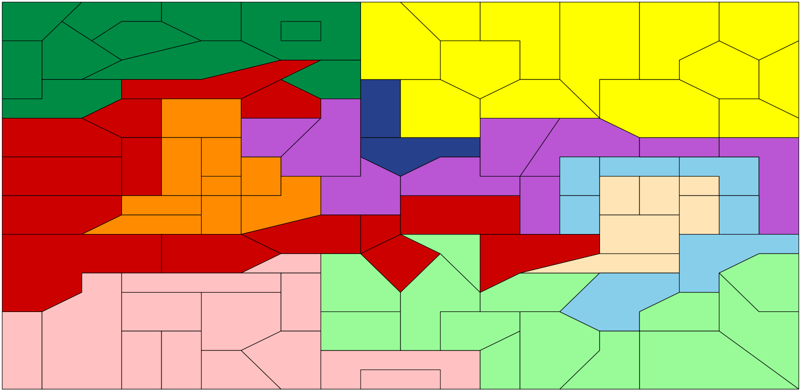
Complication due to the minimal exposure constraint



Similarity between adjacent groups



Similarity between adjacent groups



Additional step : resolving problems with small groups (1/2)

1. Define a counter $c \leftarrow 0$ and define c_{max} to be the maximum number of consecutive iterations resulting in no change in the map that can be done ;
2. Save the map in its actual form and the value of the total deviance, say D_0, which is associated to it ;
3. If $c = c_{max}$, stop the algorithm ; if not, choose an exposure threshold, say e_s, determining which groups are considered small ;
4. Merge the small groups, that is the groups with an exposure less than e_s, to their most similar neighbor ;

Additional step : resolving problems with small groups (2/2)

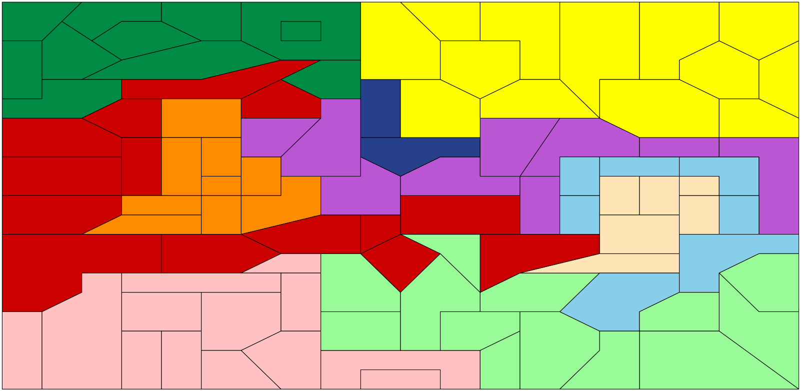
5. Apply the modified k-means algorithm restraining the database to a particular group and producing only two groups (within the chosen group) ; replicate for all the remaining groups and make the move which is the most profitable ; optimize the map ; repeat until there are k groups ;

6. Compute the new deviance, say D_1 ; if $D_1 < D_0$, let $c \leftarrow 0$ and go to step 2 ; otherwise, let $c \leftarrow c + 1$, replace the actual map by the one saved in 2 and go to step 3 ;

Example 1 (cont'd)

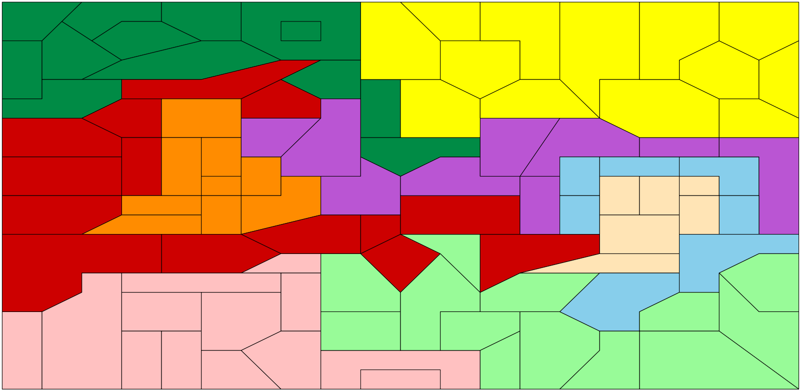
Additional step

Example 1 (cont'd)



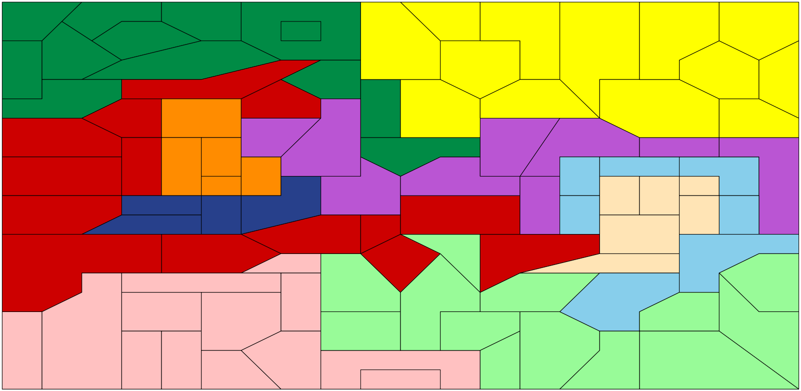
$e_s = 10,408$ # of groups available : 0 $D = 69,066,748$

Example 1 (cont'd)



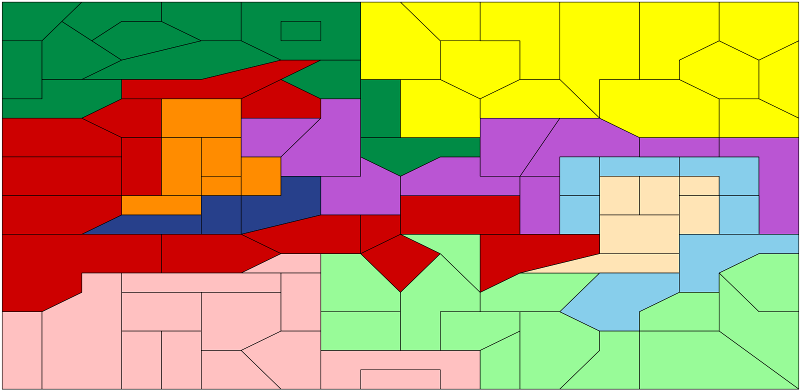
$e_s = 10,408$ # of groups available : 1

Example 1 (cont'd)



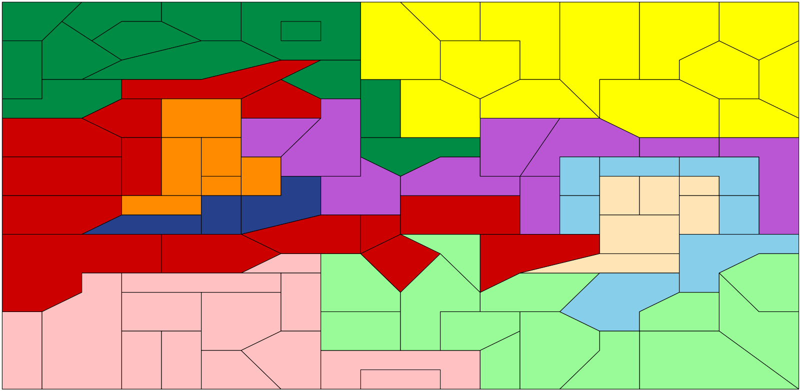
$e_s = 10,408$ # of groups available : 0

Example 1 (cont'd)



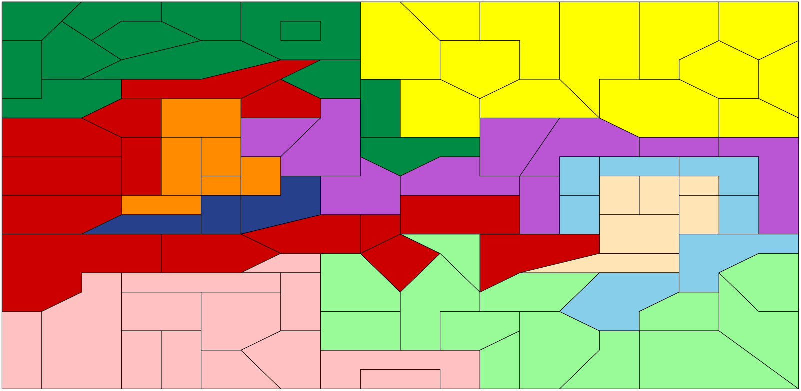
$e_s = 10,408$ # of groups available : 0 $D = 59,634,652$

Example 1 (cont'd)



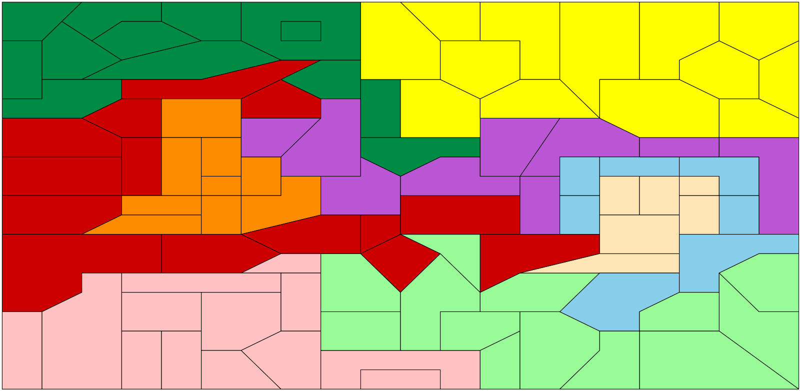
$e_s = 11,177$ # of groups available : 0 $D = 59,634,652$

Example 1 (cont'd)



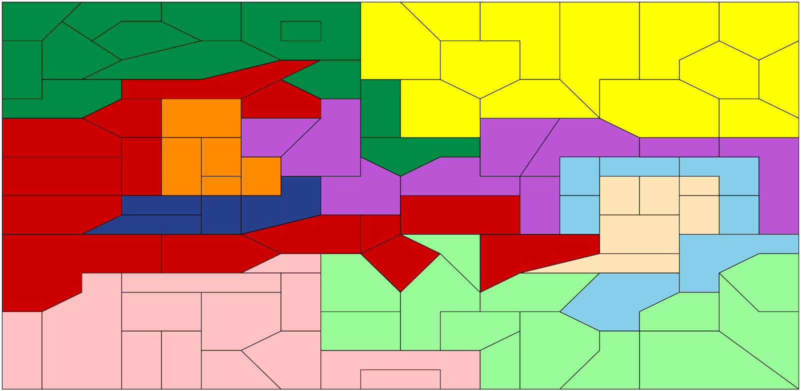
$e_s = 17,203$ $\quad$ # of groups available : 0 $\quad$ $D = 59,634,652$

Example 1 (cont'd)



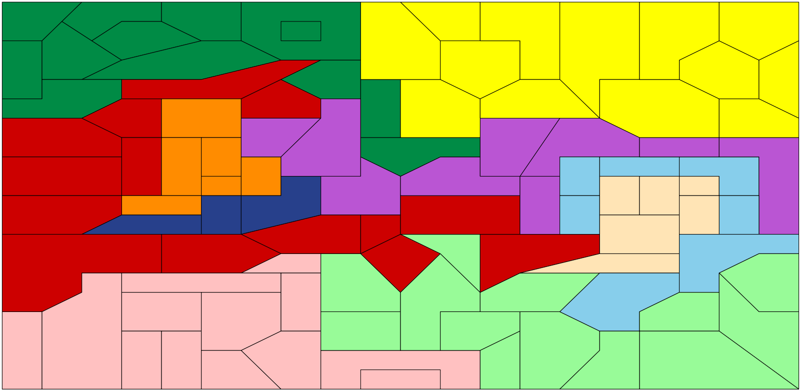
$e_s = 17,203$ # of groups available : 1

Example 1 (cont'd)



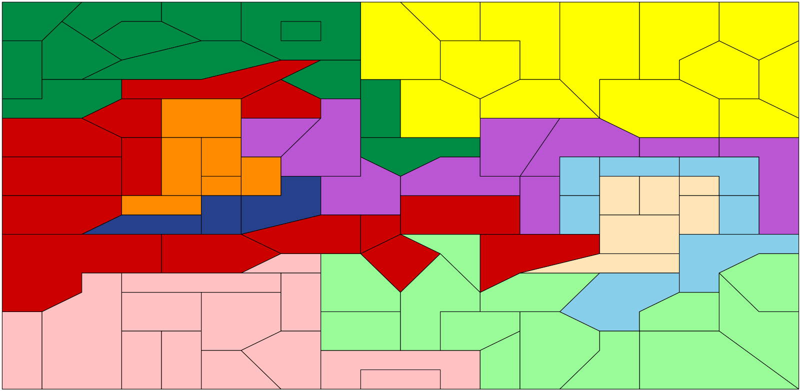
$e_s = 17,203$ # of groups available : 0

Example 1 (cont'd)



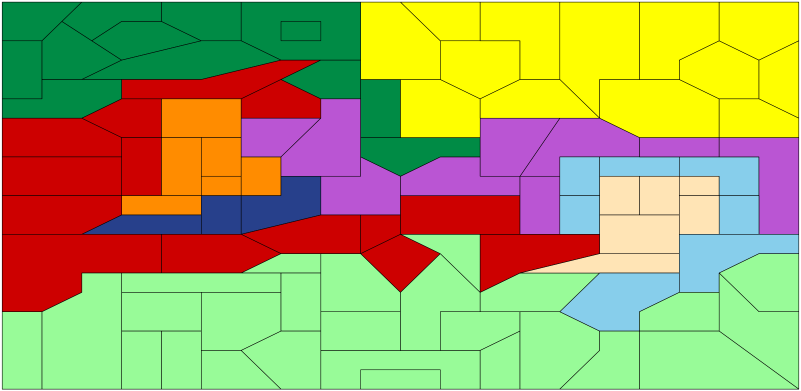
$e_s = 17,203$ # of groups available : 0 $D = 59,634,652$

Example 1 (cont'd)



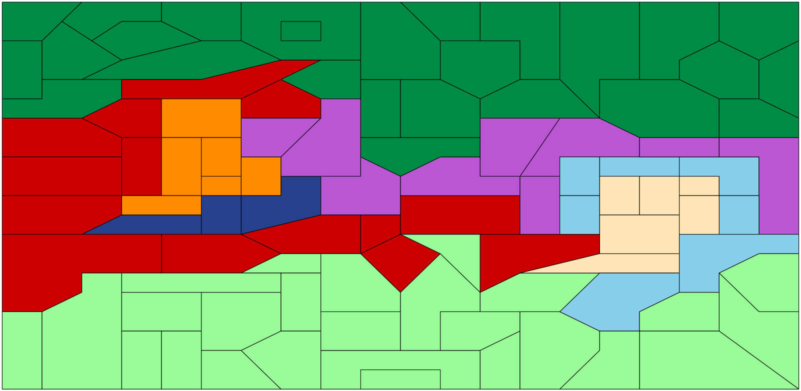
$e_s = 25,985$ # of groups available : 0 $D = 59,634,652$

Example 1 (cont'd)



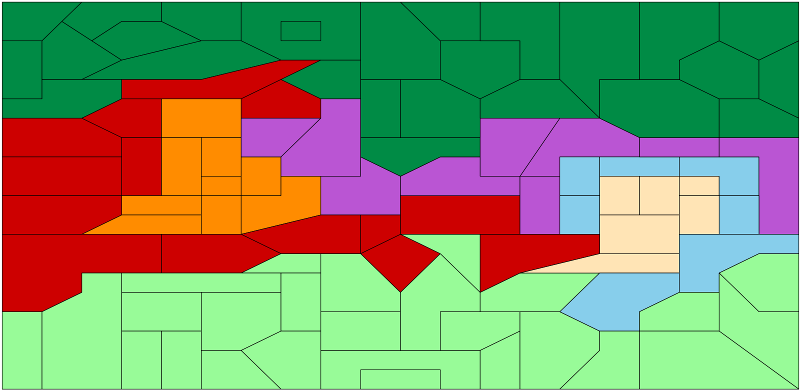
$e_s = 25,985$ # of groups available : 1

Example 1 (cont'd)



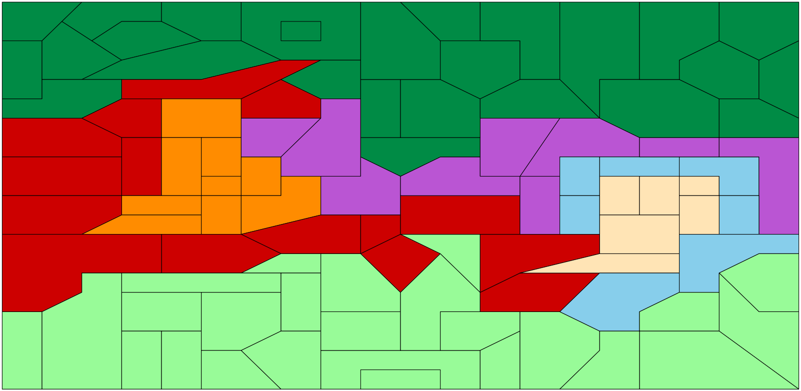
$e_s = 25,985$ # of groups available : 2

Example 1 (cont'd)



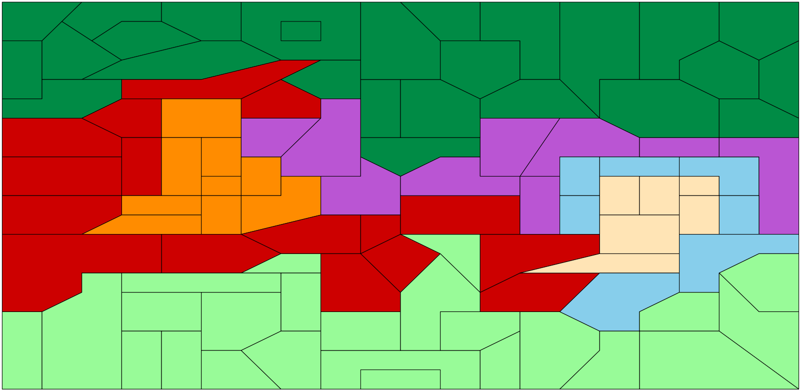
$e_s = 25,985$ # of groups available : 3

Example 1 (cont'd)



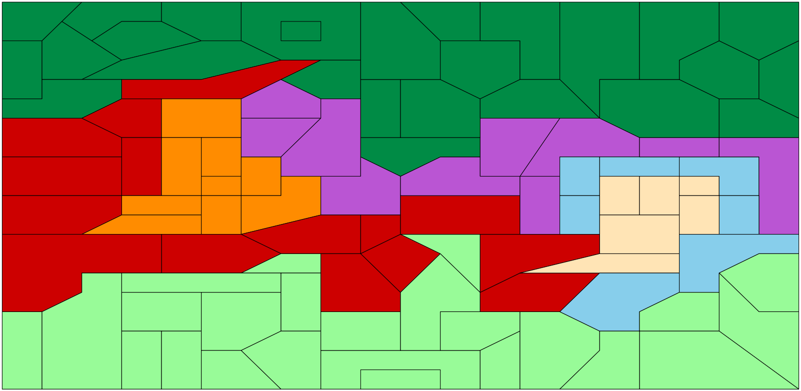
$e_s = 25,985$ # of groups available : 3

Example 1 (cont'd)



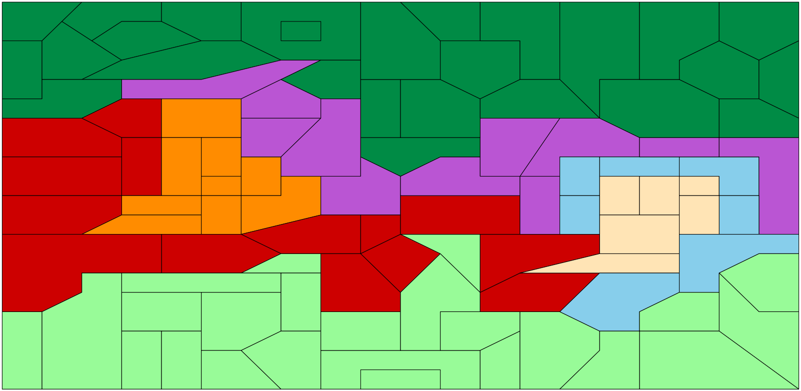
$e_s = 25,985$ # of groups available : 3

Example 1 (cont'd)



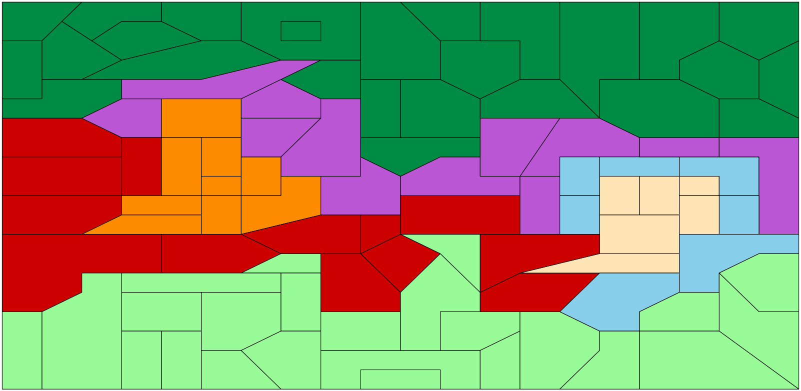
$e_s = 25,985$ # of groups available : 3

Example 1 (cont'd)



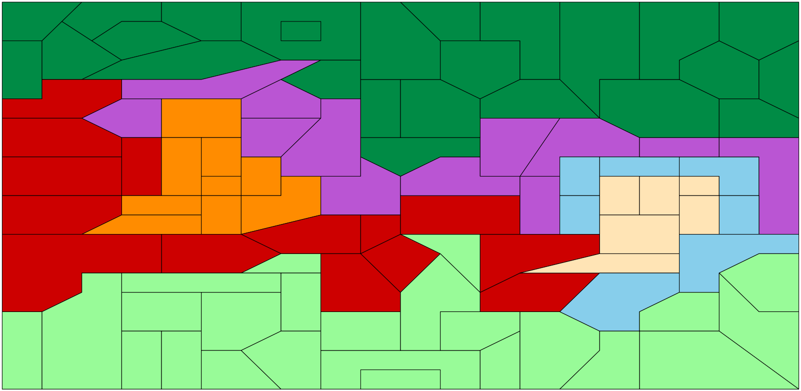
$e_s = 25,985$ # of groups available : 3

Example 1 (cont'd)



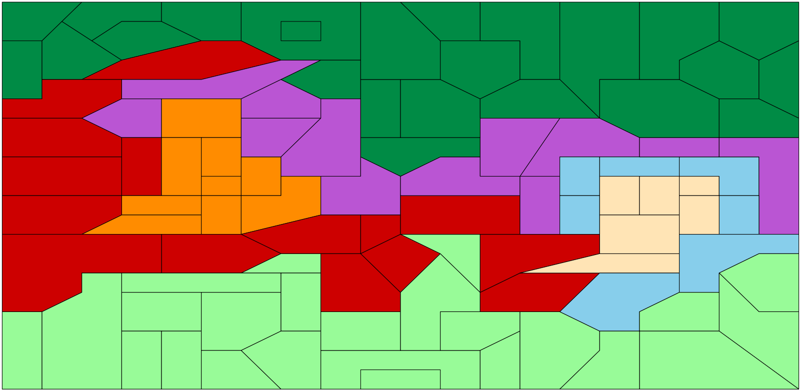
$e_s = 25,985$ # of groups available : 3

Example 1 (cont'd)



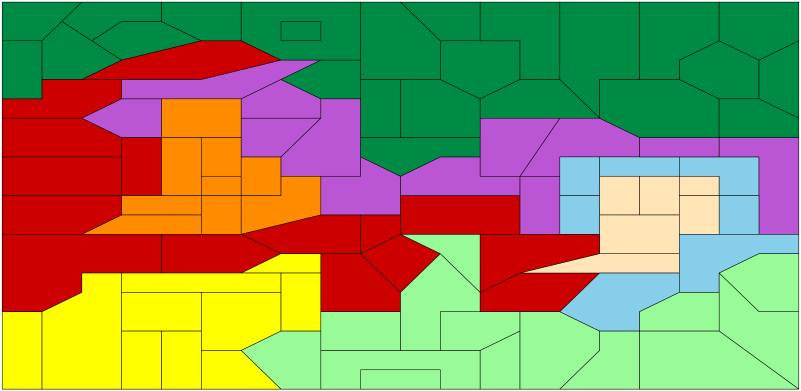
$e_s = 25,985$ # of groups available : 3

Example 1 (cont'd)



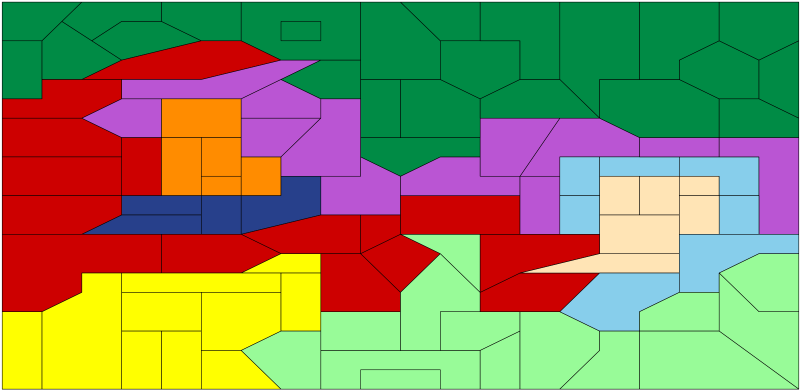
$e_s = 25,985$ # of groups available : 3

Example 1 (cont'd)



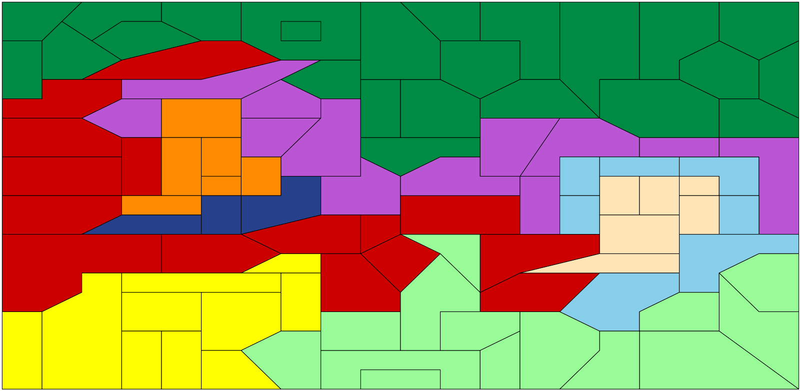
$e_s = 25,985$ # of groups available : 2

Example 1 (cont'd)



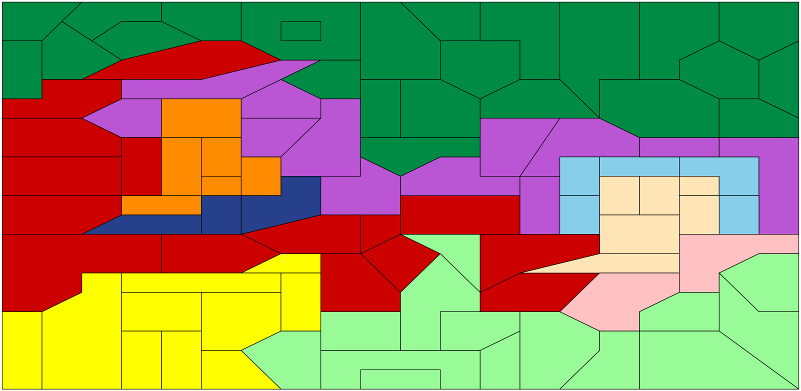
$e_s = 25,985$ $\qquad$ # of groups available : 1

Example 1 (cont'd)



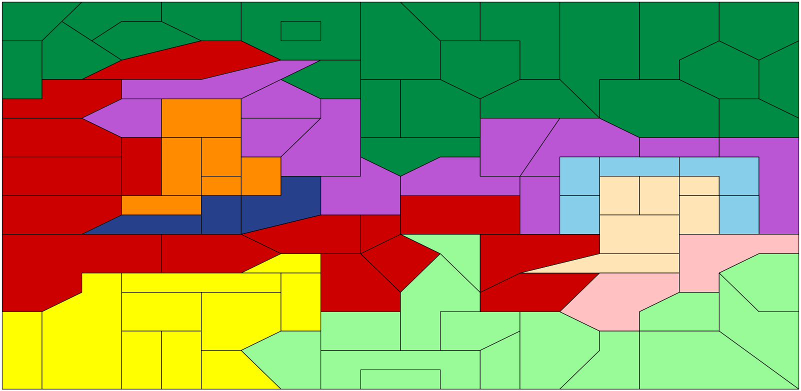
$e_s = 25,985$ # of groups available : 1

Example 1 (cont'd)



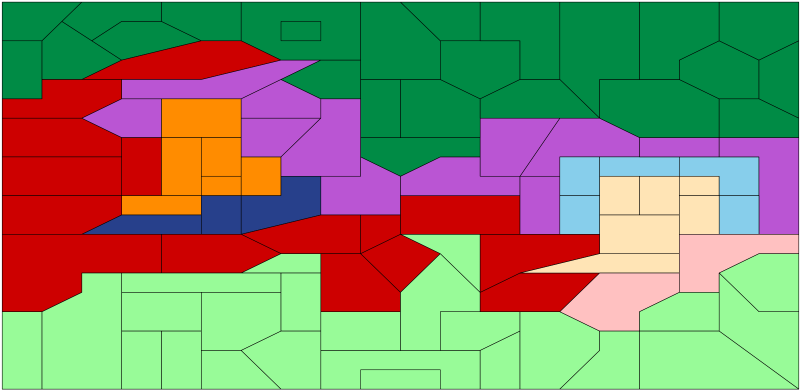
$e_s = 25,985$ # of groups available : 0 $D = 47,877,278$

Example 1 (cont'd)



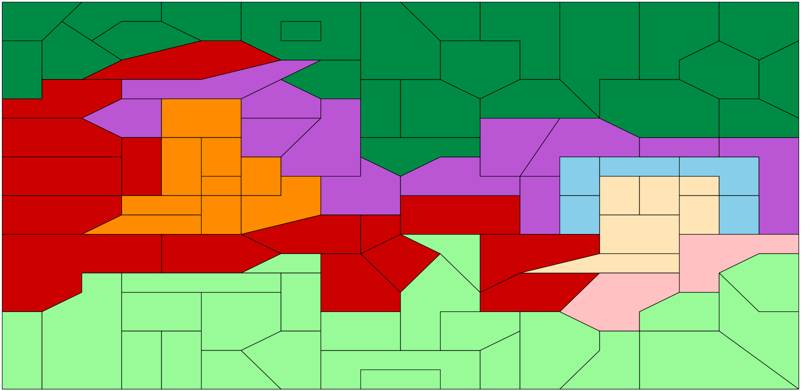
$e_s = 24,692$ # of groups available : 0 $D = 47,877,278$

Example 1 (cont'd)



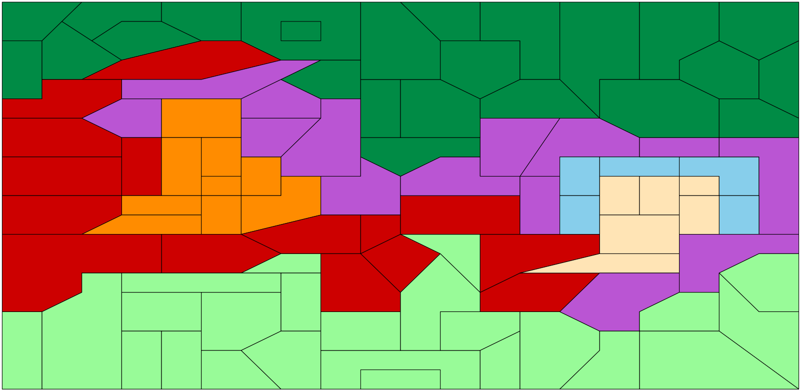
$e_s = 24,692$ # of groups available : 1

Example 1 (cont'd)



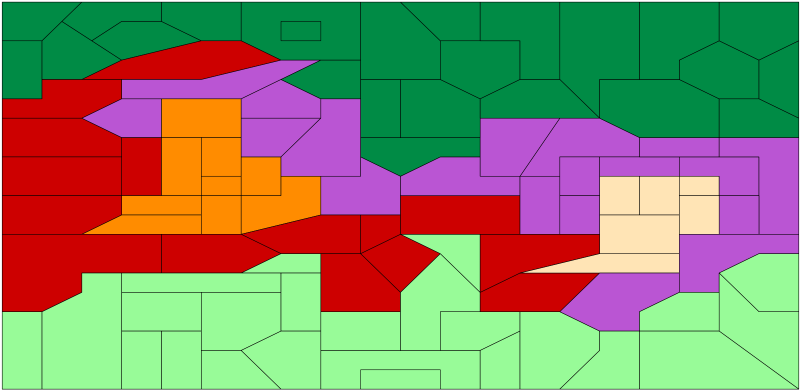
$e_s = 24,692$ # of groups available : 2

Example 1 (cont'd)



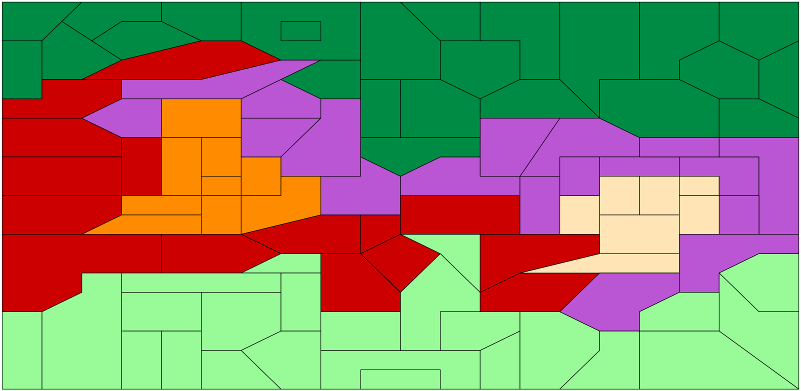
$e_s = 24,692$ # of groups available : 3

Example 1 (cont'd)



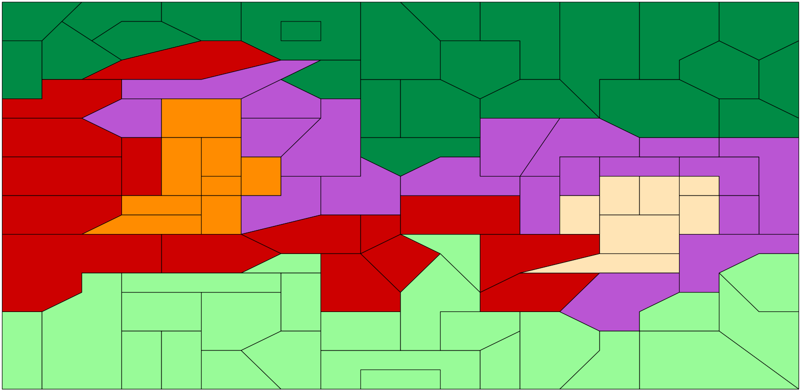
$e_s = 24,692$ # of groups available : 4

Example 1 (cont'd)



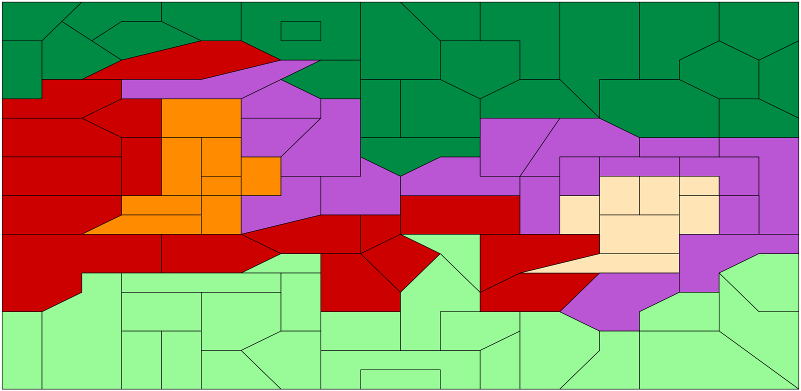
$e_s = 24,692$ # of groups available : 4

Example 1 (cont'd)



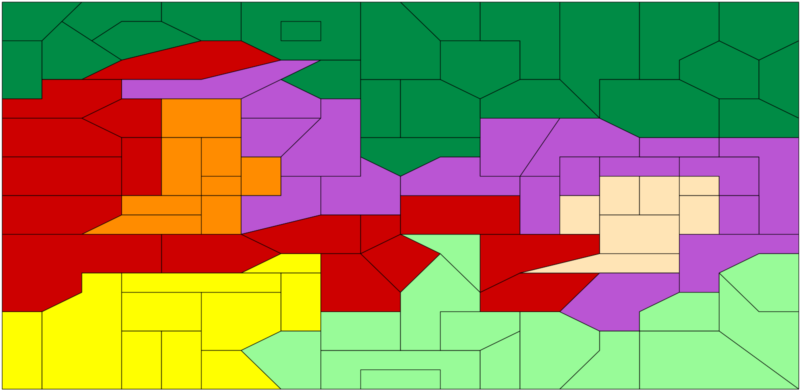
$e_s = 24,692$ # of groups available : 4

Example 1 (cont'd)



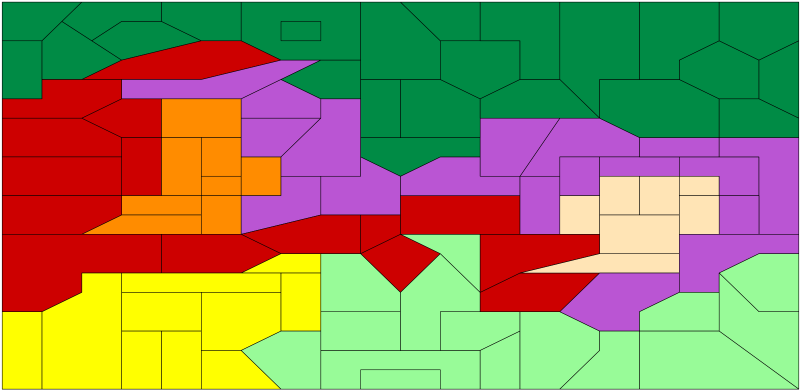
$e_s = 24,692$ # of groups available : 4

Example 1 (cont'd)



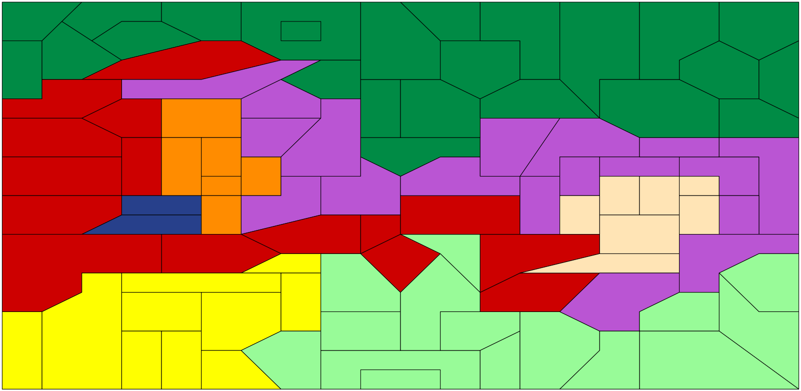
$e_s = 24,692$ # of groups available : 3

Example 1 (cont'd)



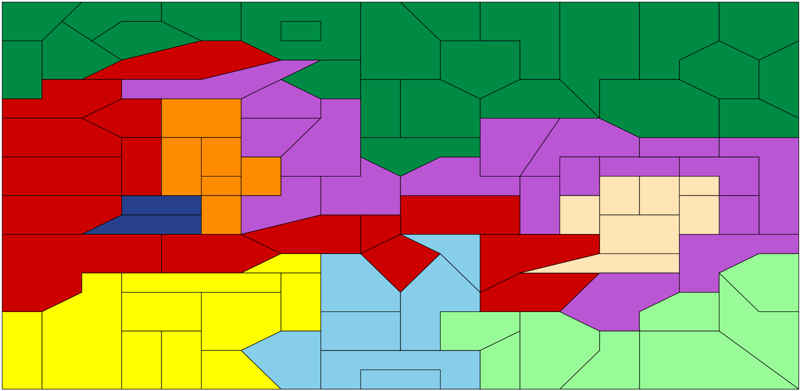
$e_s = 24,692$ # of groups available : 3

Example 1 (cont'd)



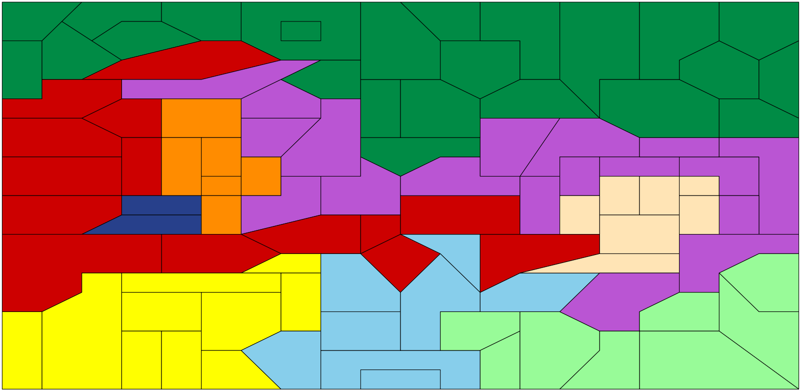
$e_s = 24,692$ # of groups available : 2

Example 1 (cont'd)



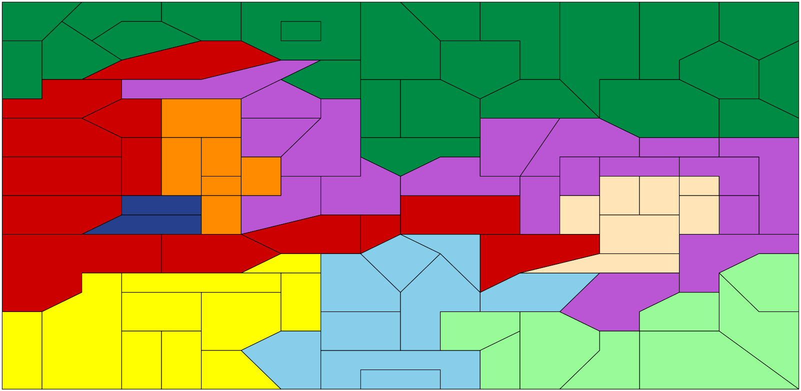
$e_s = 24,692$ # of groups available : 1

Example 1 (cont'd)



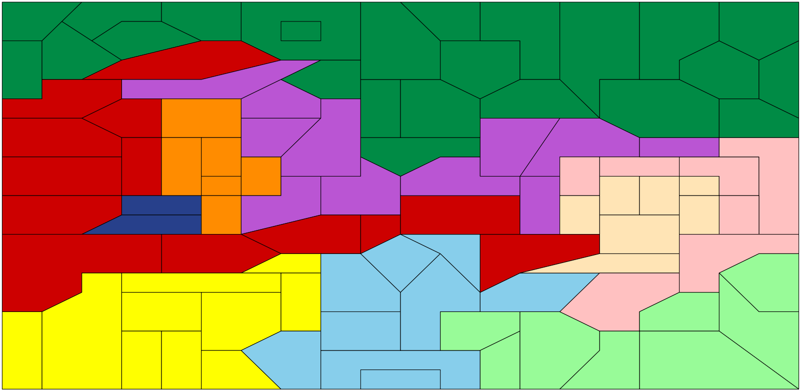
$e_s = 24,692$ # of groups available : 1

Example 1 (cont'd)



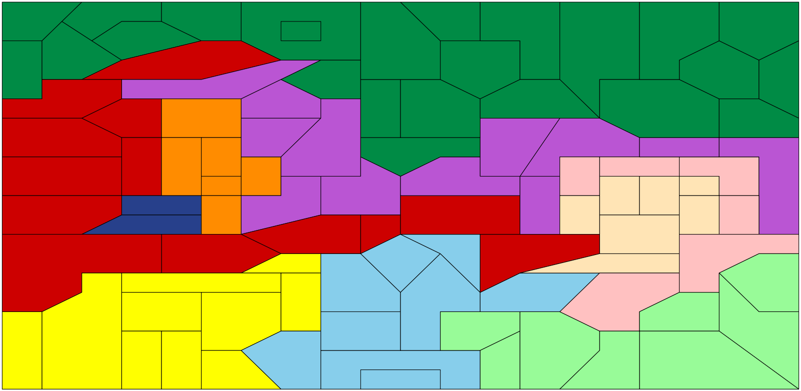
$e_s = 24,692$ # of groups available : 1

Example 1 (cont'd)



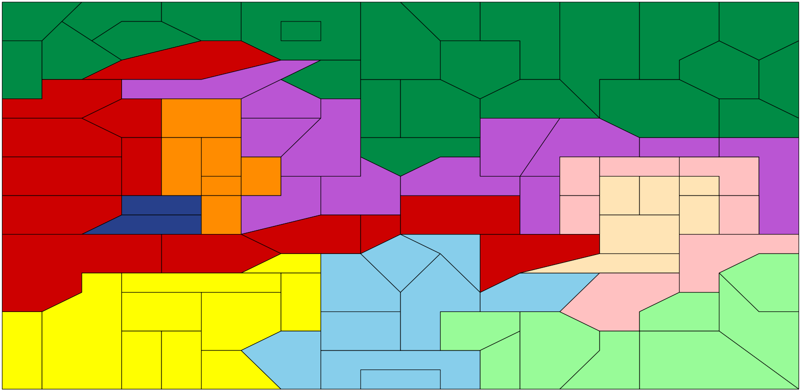
$e_s = 24,692$ # of groups available : 0

Example 1 (cont'd)



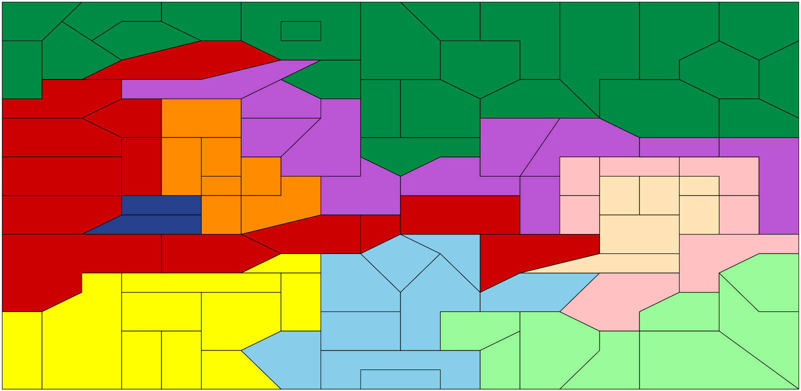
$e_s = 24,692$ # of groups available : 0

Example 1 (cont'd)



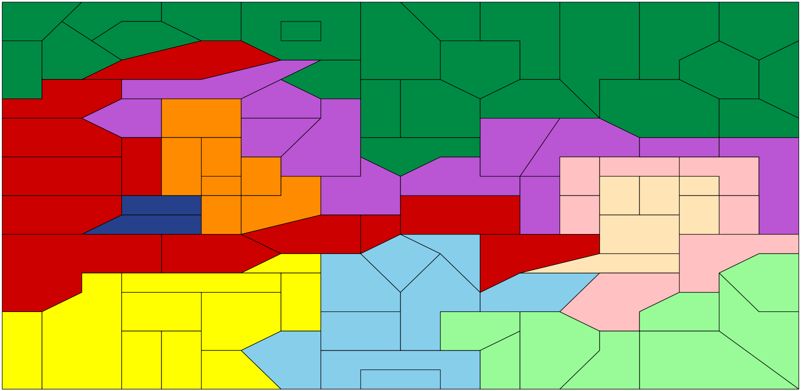
$e_s = 24,692$ # of groups available : 0

Example 1 (cont'd)



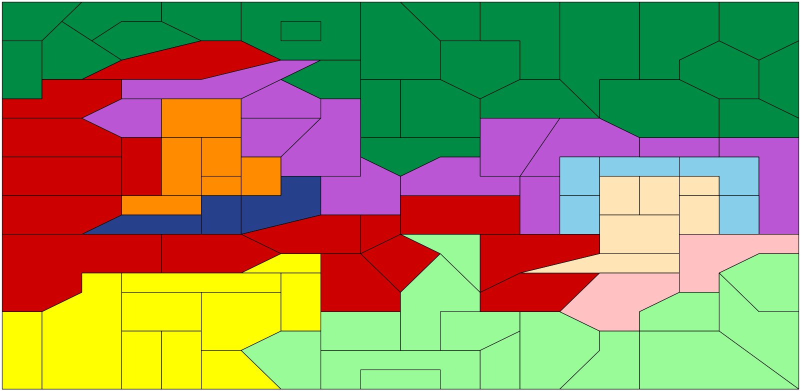
$e_s = 24,692$ # of groups available : 0

Example 1 (cont'd)



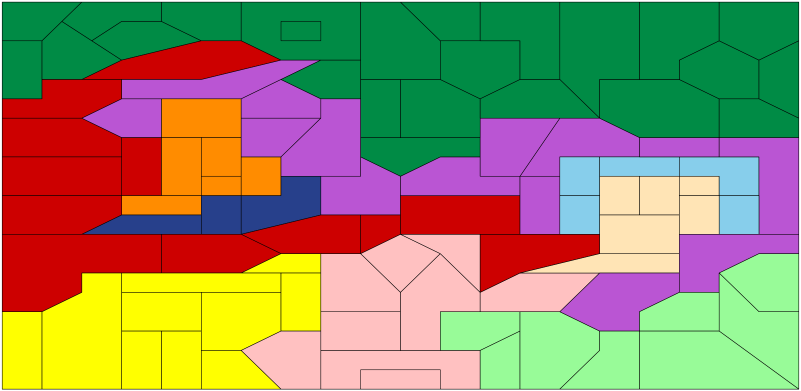
$e_s = 24,692$ # of groups available : 0 $D = 58,198,865$

Example 1 (cont'd)



$e_s = 21,284$ # of groups available : 0 $D = 47,877,278$

Example 1 (cont'd)



$e_s = 16,726$ # of groups available : 0 $D = 45,342,835$

Potential improvement

One could merge adjacent territories that are similar.

Partly from [MacQueen, 1967]

- Part of a "k-means algorithm" allowing k to vary.
- The idea is to compare the means.
- Let $c_{\min}$ be the maximal distance between two groups for them to be considered similar. Two groups are merged together if the distance between their means is less than $c_{\min}$, implying that k diminishes.

Section 5

Dealing with the constraint specific to Toronto

Separate application of the algorithm + final optimization

Separate application of the algorithm

In order to deal with the restriction on the Toronto area (10 final groups), the database is split into two disjoint databases.

Final optimization on the entire database

Additional condition : no new group can enter the Toronto area.
$\rightarrow$ It is allowed for a group already in Toronto to expand outside of it.

Alternative

The algorithm could have been constructed so that no separate application were necessary. (Worth it ?)

Section 6

Application and results

The database

Ontario

- 524 territories with 102 in the area of Toronto
- Minimum exposure : 0
- Maximum exposure : 23,736

Reminder of the constraints

- Maximum number of territories allowed : 55
- Maximum number of territories allowed in Toronto : 10
- Minimum exposure required in (final) territories : 7500

Note : Exposures were collected over a three-year span.

Results

	Without the add. step			With the add. step		
Region	Toronto	Else	Total	Toronto	Else	Total
Num. sim.	30,000	30,000	-	2,000	200	-
Min. dev.	78	82	150	72	43	115

Note : The deviances are given in millions.

Section 7

Conclusion

The method

The goal

Get the lowest intra-group variation attainable.

Basis : the k-means algorithm

Designed to minimize intra-group variation

Modifications

- Initialization
- Reassignment
- Additional step

Disadvantage and solution

Very lengthy

- Performing one simulation is time consuming (and the database could be bigger !)
- Lots of simulations needed (really ?).

Solution proposed

The second additional step presented could be the solution.

→ Reduce the number of simulations needed.

→ Improve the result.

References I

Forgy, E. W. (1965).
Cluster analysis of multivariate data : efficiency versus interpretability of classifications.
Biometrics, 21:768–769.

Lance, G. N. et Williams, W. T. (1967).
A general theory of classificatory sorting strategies ii. clustering systems.
The computer journal, 10(3):271–277.

Lloyd, S. (1982).
Least squares quantization in pcm.
Information Theory, IEEE Transactions on, 28(2):129–137.

References II

MacQueen, J. (1967).
Some methods for classification and analysis of multivariate observations.
In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, volume 1, page 14. California, USA.

Steinley, D. (2006).
K-means clustering : a half-century synthesis.
British Journal of Mathematical and Statistical Psychology, 59(1):1–34.