

A System for Interactive Spatial Analysis via Potential Maps

Christine Plumejeaud(1), Jean-Marc Vincent(1), Claude Grasland(2), Sandro Bimonte(1), H el ene Mathian(2), Serge Guelton(1), Jo el Boulier(2), J er ome Gensel(1)

(1) Laboratoire d'Informatique de Grenoble, BP 72, 38402 Saint-Martin d'H eres, France, {firstname.name}@imag.fr

(2) UMR Laboratoire G eographie-Cit es, 13 rue du four, 75006 Paris, France, {firstname.name}@parisgeo.cnrs.fr

Abstract. This paper presents a new cartographic tool for spatial analysis of social data, using the potential smoothing method [6]. The purpose of this method is to view the spreading of a phenomenon (demographic, economical, social, etc.) in a continuous way, at a macroscopic scale, from data sampled on administrative areas. We aim to offer an interactive tool, accessible through the Web, but guarantying the confidentiality of data. The biggest difficulty is induced by the high complexity of the calculus, dealing with a great amount of data. A distributed architecture is proposed: map computation is made on server-side, using particular optimization techniques, whereas map visualization and parameterisation of the analysis are done on a web-based client, the two parts communicating through a Web protocol.

Keywords: multiscalar spatial analysis, potential maps, interactive maps, spatial decision support system

1 Introduction

Recent advances in Web domain have led to new research issues in interactive and dynamic cartography (Cartographic Web) [9]. In this context, the interdisciplinary research group *HyperCarte* which gathers researchers in Geography, Statistics and Computer Science works on the design and development of a set of interactive spatial analysis tools for the representation and the analysis of social, economic and environmental phenomena. Cartographic Web technologies represent a possible approach as they grant flexibility and interactivity. Indeed, spatial analysis consists in exploration spatial data in order to formulate, compare and validate hypotheses. The strong link between interactivity, exploration and data analysis is described by MacEachren [13] through the "map-use cube" which represents visually the interactivity degree, the type of target and the degree of data knowledge necessary for the user in the *Exploration, Analysis, Synthesize and Presentation* steps of the spatial decision making process. A map is interactive if it gives access to other data [1, 11]. For instance, through a simple click on a part of an interactive map, a new piece of information (another interactive map, a multimedia document, etc.) can be accessed.

There is no complex query language to be mastered by the user. Thus, interactivity enlarge the public of user, allowing non-computer science aware people, such as policy makers, geographers, statisticians, stake holders, and so on, to create potential maps.

This paper describes *HyperSmooth* a tool based on the Web, which generates dynamically continuous interactive maps using administrative, environmental, or economical data, collected on grids or territorial meshes. More specifically, the goal is to visualize and analyze the spatial distribution of various phenomena at a macroscopic level, via the usage of a method which provides a cartographic representation abstracting real observed data. HyperSmooth implements a method, called *potential transformation method* [6], which maintains the total mass of data and supplies a correct cartographic representation. So far, no Geographic Information System (GIS) [12] and Exploratory Spatial Analysis tool [7] allow computing and interactively analyzing of true potential maps. The main issues of this spatial analysis method and of its cartographic representation are the high computational cost it induces, which generally hampers interactivity. To overcome this problem, HyperSmooth distributes the computation on a multiprocessor server, which performs a parallelization of calculation tasks, making possible the visualization of maps on a web interactive client with a secured connection.

The paper is structured as follows. The potential transformation method is described in section 2. Our prototype is presented in section 3, and section 4 shows the results of our approach. Conclusion and future work are detailed in section 5.

2 Potential Transformation Method

For the purpose of spatial analysis, geographers need to give a continuous representation of data (currently collected on grids), in order to abstract from the initial grid built during the collect of these data. The potential method is an innovative approach for such problem, proposed by the HyperCarte research group [10]. This section presents the principles of the potential transformation method highlighting the complexity of its implementation, but also the great opportunities that it offers for an explorative cartography of the spreading of socio-economic phenomenon's.

A continuous cartographic representation of discrete spatial phenomena is necessary when an abstraction of the spatial grid is needed because the data grid is heterogeneous, or because the cell level has simply no meaning regarding the analyzed phenomenon. This allows for a spatial distribution of the phenomenon without any reference to the underlying administrative territorial subdivision.

The method handles the geographic space through a grid composed of territorial units. Different kinds of indicators are associated with each cell of the grid: number of inhabitants, plants, or cars, the quantity of wealth expressed by the GDP, etc. The territorial grid can be, for instance, the municipality territorial subdivision, and this sort of grid is usually nested in a grid of upper level (like the departmental grid), that is to say that cells of the lower grid are grouped to form a bigger cell in the upper

level; then the sum of the values of an indicator associated to the lowest cells is equal to the value of the formed cell in the superior grid (this is the additive property of used indicators). For each indicator, the method calculates the value of the potential in each point (or cell) of the discretized space. Discretization is a subdivision of space into regular plots using, for instance, a grid projection into a geometric space. In every location of the geometric space, the calculated potential must be understood as the likely value for the considered indicator, depending on the contribution of each cell of the geographic space weighted by the distance to this location.

Let A the set of territorial units, a an element of this set, S_a the indicator value associated to the unit. Knowing that indicators are additive, and that their contribution is proportional to the distance δ between a and the point M , we define the potential $\Phi(M)$ for a point M of the geometric space as:

$$\Phi(M) = \sum_{a \in A} S_a f(\delta(a, M)). \quad [1]$$

For instance, if A is a set of European municipalities, and S the number of centenarians living there, we wish to estimate the potential value of centenarians in a every location M of the European space. For each city a , S_a is the number of centenary inhabitants, and g_a is the centre of the municipality (i.e. the centroid, its administrative or economic centre, etc.). The distance $\delta(a, M)$ is defined as the distance d existing between a location M and g_a . Then, the contribution of each element a of A to the potential value in M is weighted by a function f of the distance d , because the effect of an indicator usually decreases with the distance: it is maximal at a null distance, and null at an infinite distance. To define correctly the potential in function of a parameter, normalization in each point O of the space is applied using [2]:

$$\int_{R^2} f(d(O, M)).dM = 1, \text{ is also } \int_A \Phi(M).dM = \sum_{a \in A} S_a. \quad [2]$$

The total sum of the indicators is equal to the integral of the potential. We obtain a redistribution of the mass of the considered space. Using the gravity model metaphor, $\Phi(M)$ can be interpreted as the attraction of the environment on a mobile point situated in M , whose displacement vector is $-grad \Phi$. Another possible interpretation is that $\Phi(g_a)$ measures the influence of a mass situated at point g_a on the set of points M of its neighbourhood. By example the centenarians living in Nuoro, a city of the Sardinia region, will contribute much more to the estimation of a point M situated in its neighbourhood than the centenarians living in Roma. From a methodological point of view, this approach is quite similar to signal treatment processing based on the deconvolution of the sampled signal. The computational cost of the equation [1] depends on the administrative grid size (the number n of elements of A) and on the resolution of the image that represents the map (the number m of estimated points M).

The calculation depends mainly on the function f , called *spatial interaction function*. The shape of the function integrates the hypothesis made about the modalities of the spatial diffusion of the studied phenomenon. Three models of

functions are proposed: a model with a limited support (*Disk*), an exponential model for near interactions (*Gaussian*), and a model with a long scope (*Exponential*). The last method allows, for instance, to model and study the propagation of the human epidemics: their diffusion could be represented on long or short distances, according to the mobility range of the contaminated element (human, animal or plant), depending on its conveyance. User can test the different models applying different functions. The analysis of the phenomenon depends also on the scope p of the interaction function. The scope is defined as the average distance action of a mass on its neighbourhood. It is linked to the form of the interaction function by the equation

$$p = \int_{R^2} d(O, M) f(d(O, M)) . dM = \int_0^{+\infty} f(r) 2\pi r^2 . dr . \quad [3]$$

The scope could be interpreted as the spatial scale of the chosen study for the representation. The couple (function, scope) supports the economic and sociologic hypothesis regarding the spatial diffusion of the studied phenomenon. For an exploratory approach, the ability to choose the values of this couple (function, scope) is crucial for the user, who would like to modify interactively those two parameters according to the resulting map. For instance, the figures 2 and 3 shows two potential maps representing the population density, made with Gaussian function on scope of 250 km and 1000 km respectively. By comparison, the map of figure 2 is more detailed and shows the high-density population areas more precisely than the map of the figure 3 that, on the contrary, generalises the phenomenon.

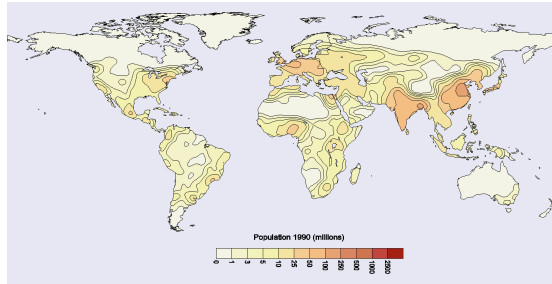


Fig. 1. Potential map of population in the World, Gaussian function, with a scope of 250 km

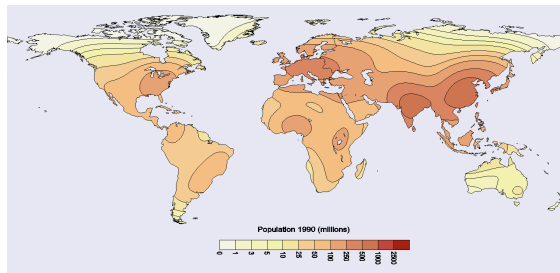


Fig. 2. Potential map of population in the World, Gaussian function, with a scope of 1000 km

At last, the type of the distance used for computation is not trivial. For instance, it can depend on the size of the space covered by the map: at a continental scale, it is not possible to use the Euclidian distance without introducing bias. The orthodromic distance is then a better choice because it takes into account the sphericity of the planet. But also, the analyzed phenomenon could necessitate other distances taking into account the anisotropy of the space (i.e. the transport distance, etc.).

3 Research Motivation

Geographic Information Systems (GIS) allow storing, visualizing and analyzing spatial data [12]. Spatial data can then be analyzed by means of geostatistic and classical analysis tools. Commercial GIS implement several vectorial data analysis tool (i.e. buffer, overlay, etc.). Some interpolation methods to create continuous maps (i.e. Inverse Distance Weighted (IDW), Krigging, Spline Polynomial Trend) have also been implemented. But it has been recognized that GIS are not Spatial Decision Support Systems (SDSS) [8]. SDSS helps decision makers to solve spatial decisional problems by providing a simple, interactive and flexible interface, managing aggregated data, handling complex spatial data structures (i.e. spatial hierarchies, field data, networks, etc.), and granting effective response times. Different kinds of SDSS have been developed. For instance, Visual Spatial Data Mining systems [4] integrate spatial data mining and GIS functionalities, Spatial OLAP tools add GIS functionalities to OLAP systems [3], and Exploratory Spatial Data Analysis tool (ESDA) [7] systems allow exploring vectorial and field data by means of interactive maps and graphic displays (scatter plots, histograms, parallel coordinates). In those systems, user can interact with the map and trigger spatial analysis operators. Spatial analysis tools for field data implemented in SDSS systems offer many functionalities such as: summarization (i.e. data is aggregated for each cell of the grid map), reclassification (i.e. data value is transformed using data mining algorithms and or statistical methods, etc.), change of resolution of a grid, user-defined weighted point transformations, triangulation or detection of polygons, and calculation of weighted densities [2]. SDSS systems provide adequate user interfaces to provide the best insight into the spatial data set, as well as particular computation methods to speed-up calculation times. However, to best of our knowledge no GIS nor ESDA implement the potential method. In order to explore and analyze geographic phenomenon at multi-scales using this innovative method, there is a need for an ESDA system implementing the potential method, providing a user-friendly interactive interface. Due to the high cost of the computation, this system should ensure high calculus performances. The system should also protect the user data.

3 Design of an Interactive Web-based System computing Potential Maps

In this section, we present our prototype, *HyperSmooth*, for the calculation and the visualization of potential maps. *HyperSmooth* permits, through particular optimization techniques, to rapidly compute potential (continuous) maps, and thus offers a high degree of interactivity, which is required for spatial analysis.

HyperSmooth is based on a client-server architecture with a Web-based client and a cluster of computation stations (see figure 4). The computation of potential maps requires a huge volume of processing resources. The Java client achieves the calculations for the cartographic visualization, while the server is responsible for the heavy computation of potential maps. At this stage, the computation only uses the orthodromic distance.

The server optimizes the data management in order to produce intermediary results in very few seconds which guaranties interactivity. In the following subsections, we detail the communication between the client and the server, the optimization techniques, and the visual interactive user-interface.

The server is accessible via the SOAP protocol (*Simple Object Access Protocol*) with a secure connection using SSL (*Secure Socket Layer*) securing confidential data.

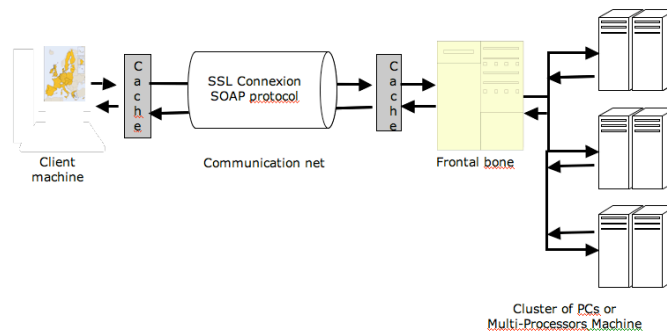


Fig. 4. The distributed architecture of HyperSmooth.

3.1 Client-Server Communications

Security is mandatory for our application because not all users have grants to visualize potential maps, and it is possible to calculate original confidential data using potential maps. Then the communication between the server and the client is realized by means of the protocol SOAP [15] for its security, portability and accessibility properties. Indeed, SOAP can use HTTP and HTTPS ports, which usually are not filtered, and it can be coupled with SSL, which provides the encryption and security. Thanks to HTTP(S), users can get a secured access on the service anywhere at anytime. Moreover, Web services grant interoperability to the server side, making

HyperSmooth an extensible framework where new GIS or SDSS clients could be plugged on the web service offered by the server side.

The potentials grids calculated by the server depend on various parameters (the resolution, the framing, the interaction function and the scope), that are specified by the client to the server. The exchanges between the two parts are limited to those elements, and then the client manipulates the grid data to build the image. Indeed, the client is in charge of the customisation of the map to comply with user's preferences, such as the color pallet, the number of classes and the type of the progression for the distribution of the colors. The client can generate reports (text and/or HTML files) on the fly. These reports contain the geographic coordinates of each point M , together with its potential value. The client saves these files (that is to say the grid data, including the calculus parameters) allowing to rapidly repainting the image when the graphic device is changed.

3.2. Optimization Techniques

Caching techniques are not adapted to the potential problem because each query has to generate a global result, which cannot be pre-calculated as it depends on the parameters of the analysis. However, a detailed analysis of the calculation tasks performed on the server's side shows that there are redundancies which can be avoided to optimize some parts of the calculation. Indeed, two problems arise when computing the potential $\Phi(M)$ (cf. equation [1]): the sum is applied to a huge number of elements, and the cost of the orthodromic distances $d(M, ge)$ calculus is high because it requires the computation of arccosine, cosine and sine angles.

The sum is reduced using a *cut-off* algebraic method. Contrary to the geometric cut-off which reduces the calculation to a fixed radius, our method takes into account some distant points whose weight S_a (statistic value) acts upon the result of the calculation. The cut-off algorithm exploits the organisation of data into a quad-tree. A point M with its coordinates and its indicators are associated to each leaf of this tree. The depth of the tree is n . Each point is then summed by groups of 4 elements (the neighbours in the grid) – see figure 5. The calculation of the potential $\Phi(M)$ is called recursively summing the product between the indicators associated to the leafs and their distance from the point M . The visit of each branch of the level $n-1$ depends on the following test, which checks if the weight of the children of the node $n-1$ is negligible or not:

$$\left(\sum_{node_{n-1}} S_e \right) * d_{\min} \leq \varepsilon * \Phi(M)_{summed} . \quad [4]$$

A crucial step of the algorithm is the definition of the epsilon value in order to not discard too many points. By default this value is 1/1000 of the sum of the indicators.

The evaluation of formulas with terms in arccosine, sine, cosine is slows down the calculus, but a tabulation of those functions allows to overcome this problem. Indeed, the values of those functions are pre-computed on a fine and regular subdivisions of a given interval, and then, the algorithm estimates the value of a certain angle with the

nearest pre-computed angle value. The number of subdivisions of this interval can be chosen.

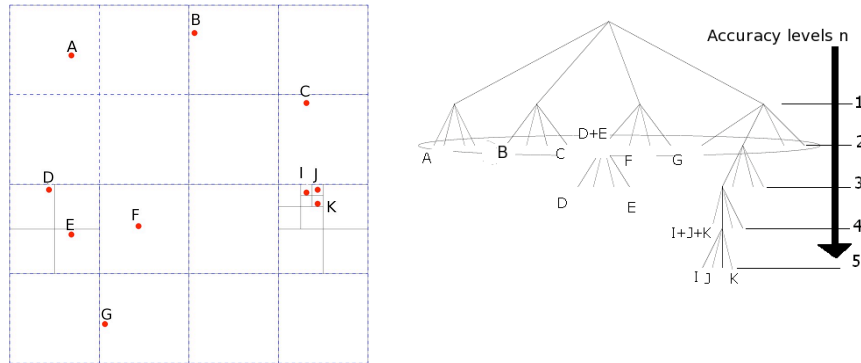


Fig. 5. An example of tree construction from a discretized study space.

3.3. Interactive Visualization

The user interface is composed of a main panel containing the interactive map (Data Panel) and several windows that permit to parameterize the spatial analysis (Control Panel). User can select the dataset he wishes to work with in the “Control Panel”, see figure 6-2. There are two combo boxes listing available datasets, one for the numerator and the other one for the denominator of the potential ratio.

The specific parameters for the computation of potential maps are: the type of the interaction function to be applied, the resolution of the computed grid and the scope. All these parameters can be selected through the panel shown on figure 6-1. There are four types of functions listed in a combo-box: *gaussian*, *exponential*, *disk* and *amortized disk*. The resolution is expressed like the number of cells in width and height of the computed grid, in a free text field. The average scope is an ordinal value expressed in kilometres, which can be entered either in a text field, or using a cursor on a slide bar.

The cartographic representation of the potential map is paint in a tab inside the Data Panel (see figure 6-5). The visualization of a potential map is obtained by clipping the raster image on the vector map of the study space. The raster image colour graduation reflects the intensity of the phenomenon, and can be customized for each map. The Data Panel contains also three panels for the colour pallet, the type of colour distribution, and the number of colour classes (see figure 6-3 and figure 6-4).

Through the Control Panel functionalities, the user can tune its spatial analysis and the results will be automatically refreshed in the Data Panel. Moreover, the client redefines classical navigation map facilities: zoom and pan. These functions trigger

the computation of potential maps whose results are quickly returned. Thus, the cartographic component is composed of interactive maps which are computed in response to user's actions in effective times allowing exploration and analysis of spatial continuous phenomena.

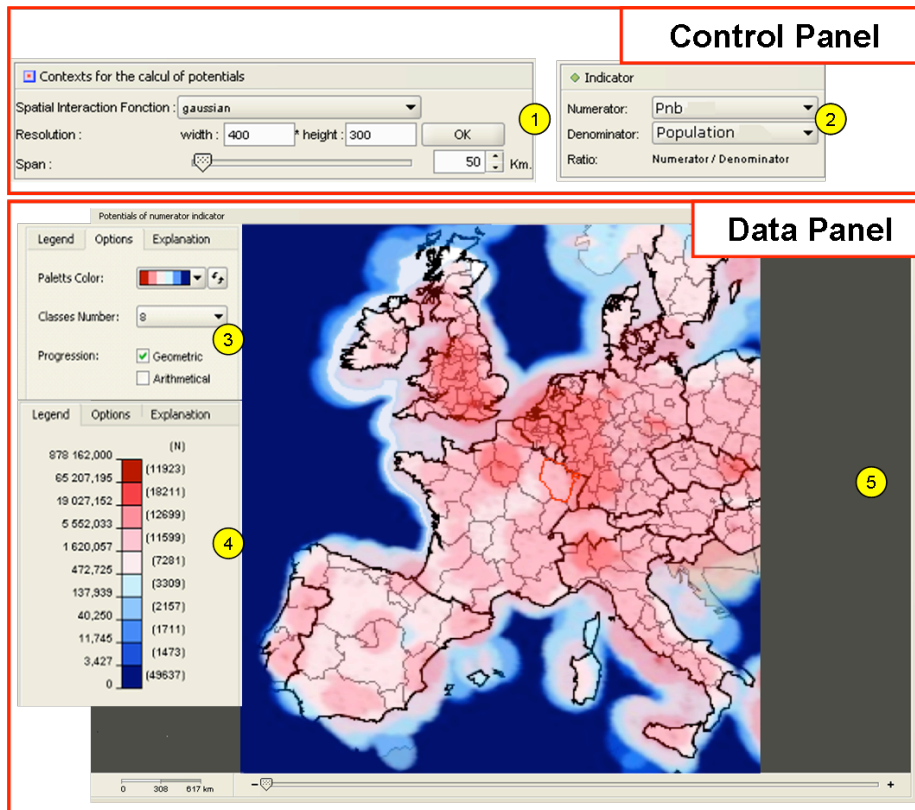


Fig. 6. The HyperSmooth user interface. Control Panel: 1) Spatial interaction function, scope and resolution, 2) Dataset to explore, Data Panel 3) Options, 4) Legend, 5) Interactive grid map.

4 Experiments: Population Distribution at different scales

In this section we present the experiments conducted on the dataset of the European population counted at municipal level, which contains 116203 geographical entities. These data have been analyzed using HyperSmooth to find a measure of the local polarization in Europe using a poor information (population and surface of cities on 1999) but very spatially detailed at cities level [6]). Experiments have been led in two different contexts: first, the server was running on a standard dedicated machine, and latter, the computation tasks was parallelized and the server was unfolded on a Shared

Multi-Processors (SMP) machine. In both cases, client could access via the secured SOAP protocol to the server. Since the server can be used independently of the client, and for avoiding any alteration of measures introduced by net latencies, the performances were measured on the server side.

5.1 Experiments on a Standalone Server

Experiments have been made on the server's side running on a bi-processor machine, equipped with Linux OS (Pentium 4 at 2,6 Ghz, with 1 Go of memory), which appears to be a very standard working machine. The complexity of the algorithm is ascertained by the computation time measures (see Table 1). This is linearly proportional to the resolution (total number n of points to be computed on the grid), to the number m of geographical entities (which is fixed in our experiments). The cost of the calculus is indirectly linked to the scope: a longer scope will involve more geographical units in the computation than a shorter one.

The complexity being known, this allows the user to estimate the duration of each map computation, according to the requested resolution and scope. This duration is quite reasonable when using standard computers: around 2 minutes for a long scope (100 km) and a thin resolution (800 x 600).

Resolution	Scope (km)	Calculation duration (s)
200x100	100	5
400x300	100	32
800x600	100	130
800 x 600	25	33
800 x 600	50	57
800 x 600	100	130

Table 1. Potential map calculation duration

The network latency times are good, even though no data compression has been applied. Indeed, the time for the creation of the answer, its encryption, and the unpacking is 4 seconds locally with a resolution of 300x400. On the client, the building of a raster image using the grid values is fast (128 ms). This matrix is recalculated when the user changes the parameters of the analysis, or when s/he resizes the study area with a zoom or a pan action. The matrix is cached with the parameters of the visual query in order to use it for future queries.

5.2 Experiments on a SMP Machine

Alternatively, the server can be unfolded on a Shared Multi-Processor (SMP) machine, shared by several users. Then the map is computed by pieces in a parallelized way by the set of requested processors.

The algorithm for calculation of a potential map iterates on all points of the grid, which are arranged in an array of size n (n is the resolution level). The parallelization of computations does not present any difficulty due to interlaced dependences, since each point can be calculated independently of others. It is based on the distribution of tasks (a task is the calculation of an entry in the table) on a number of k processors. The number of points allocated to a processor is therefore n/k , and the calculation ends up when each processor has finished calculating its portion of the table.

This naive distribution of tasks shows however two major limitations. Firstly, such an algorithm is not resistant to disturbance: if the capabilities of a processor are suddenly divided by 50%, the computing time is then extended by the time equivalent to half its task, since there is no process for automatically rebalancing workloads between processors. Secondly, our schedule is very heterogeneous: items located in areas with low density measurements are calculated much more quickly through the process of pruning, and thus some processors complete their work faster than others.

The mechanism of adaptive transmission load [14] is a solution to adapt dynamically the shared load: each "free" processor steals half of the remaining task to a busy processor. This redistribution occurs only when the busy processor completes its assigned indivisible portion of work $\alpha \log(p)$. α is a configurable parameter, which is adjusted to reduce the duration time of the lock to access the data array: this time must be much shorter than the calculation duration of the portion of the table. This concept of minimum quantity of work is introduced to avoid problems of contention when accessing the array of points: when a processor takes a task, it activates a semaphore on the array and blocks access to other processors. This inter-blocking time is short, and the definition of a optimized minimum workload avoids the frequent repetition of inter-blockages.

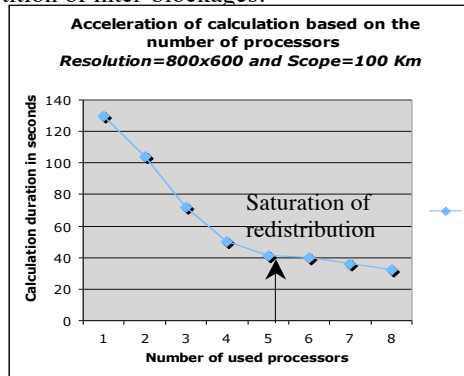


Fig. 7. Acceleration of computation depending on the number of involved processors

This algorithm has been implemented with PTHREAD in our implementation, and tested on a SMP, 8 cores with shared memory. Experiment gives results confirming the effectiveness of the method. Figure 7 show performances for a 800x600 resolution and a scope of 100 Km. Responses time pass from 103 ms for one processor to 30 ms for 8 processors. In addition, we note that this algorithm has been implemented with two different bookstores for scheduling tasks, TBB (Intel Threading Building Blocks, <http://threadingbuildingblocks.org/>) and Kaapi [5]. Whatever the bookstore, performances are improved in the same way: gain in computing time is linear with the

number of processors mobilized, as long as the redistribution is not saturated (observed from 5 processors with PTHREAD, it is postponed with Kaapi because the tasks distribution algorithm is better optimized).

6 Conclusion and Future Work

This paper presents HyperSmooth, the first system offering the possibility to interactively study the spatial propagation of social, environmental and economic phenomena through the potential transformation method. HyperSmooth is based on a client-server architecture. The Web-based client allows to visualize a set of interactive maps through a user-friendly user-interface, and drives a server which computes efficiently maps (this computation can be distributed on a grid). The client-server communication relies on the SOAP protocol and the security of data is granted by SSL cryptography. Interactivity requires short time response from the system to user's queries. For that purpose, we have developed *ad-hoc* optimization techniques, which shorten the computation duration of potential maps on the server. Calculations are speeded-up by an algebraic cut-off method and tabulation of orthodromic distances. Experiments on real dataset show good performances. Besides, the server side named Hyantes has been developed under an open-source licence, and can be downloaded on its Web site¹, allowing users to exploit and extend it for further experiments.

Currently, we are working on improving calculation performances by replacing the orthodromic distance with the euclidian one for small regions, and adopting a strategy for sub-sampling data according to the observation scale. We are also enhancing the client's functionalities by triggering potential map computation using query windows, and introducing iso-potential curves to overlay on the vectorial background map. Finally we think to couple our system with a Spatial OLAP tool in order to take advantage from its scalability, hierarchical and multidimensional data management features and usability.

References

1. Andrienko, N., Andrienko, G., Gatalsky, P. Exploratory Spatio-Temporal Visualization: an Analytical Review. *Journal of Visual Languages and Computing*, vol. 14, n° 6, pp. 503-541 (2003)
2. Andrienko, G., Andrienko, N., and Gitis, V. Interactive Maps for Visual Exploration of Grid and Vector Geodata *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 57 (5-6), pp. 380-389 (2003)
3. Bimonte, S., Wehrle, P., Tchounikine, A., Miquel, M. GeWolap: a Web Based Spatial OLAP Proposal. In: *Second International Workshop on Semantic-based Geographical*

¹ <http://hyantes.gforge.inria.fr/>

Information Systems, Montpellier, France, 29-30 October, Springer-Verlag Berlin Heidelberg (2006)

4. Compieta, P., Di Martino, S., Bertolotto, M., Ferrucci, F., and Kechadi, T. Exploratory spatio-temporal data mining and visualization. Elsevier: Journal of Visual Languages and Computing, vol. 18, n° 3 (2007)
5. Danjean, V., Gillard, R., Guelton, S., Roch, J.-L. and Roche T., Adaptive Loops with Kaapi on Multicore and Grid: Applications in Symmetric Cryptography. In : Parallel Symbolic Computation'07 (PASCO'07), ACM publishing, London, Ontario, Canada (2007)
6. Dubois, A., Gensel, J., Hanell, T., Schürmann, C., Lambert, N., Zanin, C., Ysebaert, R., Grasland C., Damsgaard O., Lähteenmäki-Smith, K., Gloersen, E., and Thomas, R., Observing the structure of European territory in relative terms *DG-IPOL, Regional Disparities and Cohesion : What Strategies for the future*, EU Parliament, report IP/B/REGI/IC/2006_201 (2007)
7. Edsall, R., Andrienko, G., Andrienko, N. and Buttenfield B., Interactive Maps for Exploring Spatial Data. Marguerite Madden (Ed.) ASPRS Manual of GIS (2008)
8. Keenan, P., Using a GIS as a DSS Generator. In : DARZENTAS John, DARZENTAS Jenny et SPYROU Thomas. Perspectives on Decision Support System. Grèce : University of the Aegean, pp. 33-40 (1996)
9. Kraak, M.-J. and Brown, A. (eds), Web Cartography, Taylor&Francis, London, (2000)
10. Grasland, C., Mathian, H., and Vincent, J.-M., Multiscalar Analysis and map generalisation of discrete social phenomena: Statistical problems and political consequences, Statistical Journal of the United Nations ECE, 17, IOS Press, pp. 1-32 (2000)
11. Josselin, D. and Fabrikant, S., Cartographie animée et interactive, Revue Internationale de Géomatique, Editions Hermès, (2003)
12. Longley, P., Goodchild, M., Mahuire, D. and Rhind, D. Geographic Information Systems and Science. New York: John Wiley & Sons, pp. 517 (2001)
13. MacEachren, A., and Kraak, M.-J. Exploratory cartographic visualization: advancing the agenda. Computers & Geosciences, vol. 23, n° 4, pp. 335-343 (1997)
14. Roch, J.-L., D. Traore, J. Bernard, Processor-oblivious parallel stream computations. 16th Euromicro International Conference on Parallel, Distributed and network-based Processing, Toulouse, France (2008)
15. W3C. SOAP. <http://www.w3.org/TR/soap/>