Neapolis University

HEPHAESTUS Repository **http://hephaestus.nup.ac.cy**

School of Information Sciences Book chapters

2015

Employing Cellular Automata for Shaping Accurate Morphology Maps

þÿUsing Scattered Data from Ro

Kapoutsis, Athanasios Ch.

Springer

http://hdl.handle.net/11728/10177

Downloaded from HEPHAESTUS Repository, Neapolis University institutional repository

Chapter 10
Employing Cellular Automata for Shaping Accurate Morphology Maps Using Scattered Data from Robotics' Missions

Athanasios Ch. Kapoutsis, Savvas A. Chatzichristofis, Georgios Ch. Sirakoulis, Lefteris Doitsidis and Elias B. Kosmatopoulos

Abstract Accurate maps are essential in the case of robot teams, so that they can operate autonomously and accomplish their tasks efficiently. In this work we present an approach which allows the generation of detailed maps, suitable for robot navigation, from a mesh of sparse points using Cellular Automata and simple evolutions rules. The entire map area can be considered as a 2D Cellular Automaton (CA) where the value at each CA cell represents the height of the ground in the corresponding coordinates. The set of measurements form the original state of the CA. The CA rules are responsible for generating the intermediate heights among the real measurements. The proposed method can automatically adjust its rules, so as to encapture local morphological attributes, using a pre-processing procedure in the set of measurements. The main advantage of the proposed approach is the ability to maintain an accurately reconstruction even in cases where the number of measurements are significant reduced. Experiments have been conducted employing data collected from two totally different real-word environments. In the first case the proposed approach is applied, so as to build a detailed map of a large unknown underwater area in Oporto, Portugal. The second case concerns data collected by a team of aerial robots in real experiments in an area near Zurich, Switzerland and is also used for the evaluation of the approach. The data collected, in the two aforementioned cases, are extracted using different kind of sensors and robots, thus demonstrating the applicability of

The research leading to these results has received funding from the European Communities Seventh Framework Programme (FP7/2007–2013) under grant agreements n. 270180 (NOPTILUS)

A.Ch. Kapoutsis (✉) · S.A. Chatzichristofis · G.Ch. Sirakoulis · E.B. Kosmatopoulos
Department of Electrical and Computer Engineering, Democritus University of Thrace, 67100 Xanthi, GR, Greece
e-mail: akapouts@ee.duth.gr

A.Ch. Kapoutsis · S.A. Chatzichristofis · G.Ch. Sirakoulis · L. Doitsidis · E.B. Kosmatopoulos
Telematics Institute, Center for Research and Technology, Hellas (ITI-CERTH), 57001 Thessaloniki, Greece

L. Doitsidis
Department of Electronic Engineering, Technological Educational Institute of Crete, 73100 Chania, GR, Greece

© Springer International Publishing Switzerland 2015

G.Ch. Sirakoulis and A. Adamatzky (eds.), *Robots and Lattice Automata*, Emergence, Complexity and Computation 13, DOI 10.1007/978-3-319-10924-4_10

our approach in different kind of devices. The proposed method outperforms the performance of other well-known methods in literature thus enabling its application for real robot navigation.

10.1 Introduction

Scattered data interpolation refers to the problem of generating the intermediate values through a non-uniform, unpredictable distribution of data samples. This numerical analysis method can be adapted in a variety of engineering fields where data is often measured or produced at random and irregular positions. The goal of interpolation is to find the best way to propagate the data, finding an underlying function [1] or utilizing the information of the neighborhood [2] and etc., onto all positions in the domain.

There are three principal sources of scattered data: measured values of physical quantities, experimental results and computational values [3]. This chapter focuses in the investigation of the first category without losing the ability of direct adjustment in other types of applications. Non-uniform measured values of physical quantities are collected in geology, meteorology, oceanography, cartography, mining, etc. Although our method applies in any of the previous categories we limit our presentation to measurement data obtained by robot teams.

In general, a key element to the successful operation of a robot team is the ability to perceive the environment in which it operates and therefore be able to function with the highest level of autonomy. Currently several types of robots including ground [4], aerial [5], surface or underwater robots [6] or even heterogeneous teams consisted of different type of robots [7] are deployed in different type of missions utilizing a diverse set of sensors. In all cases the key questions is *how the data gathered by the team members, will be processed and transformed into meaningful information, in the form of maps* so that they can be used by the robots. Usually the data collected are in the form of scattered and often noisy data.

In recent literature numerous applications of robots used for mapping of regions of interest (see e.g., Fig. 10.1) are reported. In the case of Micro Aerial Vehicles (MAVs), they have been used both in indoors [8] or outdoors [9, 10] environments using laser rangefinder sensor and a front-looking stereo camera as the main sensor respectively. A fully autonomous system using a team of MAVs has been used to construct maps of an unknown environment using a state-of-the-art visual-SLAM algorithm which tracks the pose of the camera while simultaneously and autonomously, building an incremental map of the surrounding environment [11].

In the case of underwater missions the state-of-the-art sensors, for Autonomous Underwater Vehicles and for mapping the sea-bottom, are bathymeters (sonars) or range scans [6, 12]. Furthermore, one of the sensors that is in common to all Autonomous Ground Vehicles is the sensor to perceive the environment and their movement (range sensing devices) [4].

Fig. 10.1 Real life applications of teams of Autonomous Vehicles, operating under different environment, constructed under different design architectures. **a** Unmanned Aerial Vehicles (*UAVs*) in continuous infrastructure monitoring to prevent accidents [5]. **b** Autonomous Underwater Vehicles (*AUVs*) for underwater archeology and post-disaster infrastructure inspection. **c** Autonomous Ground Vehicles (*AGVs*) during ground surveillance task

In all the aforementioned cases the vehicles' sensors produce either directly or after some processing a pool of scattered measurements of the environment in which they are operating. The elaboration of these measurements can be done either on-line or off-line. A successful demonstration of the on-line case is presented in [13] were a team of Autonomous Underwater Vehicles (AUVs) has efficiently and fully-autonomously navigated in a dynamic environment. In the off-line scenario, the robots have to follow a predefined trajectory gathering the corresponding points. After the completion of the mission the interpolation methods are used to produce the desired map. Despite the fact that, in the off-line scenario, a a-priori information

about the exact location of the measurements could be used in order to improved the performance of the interpolation process, crucial information, about the specific morphology of the area, may be lost.

There is, no unique solution to the interpolation problem, resulting in different fields when alternative techniques are applied to the same discrete data set. Since the interpolation methods in bibliography are numerous, we have used as evaluation methods four of the most common and most general applicable ones:

1. **Linear**
   Linear interpolation is the simplest method of getting values at positions between the data points. The points are simply joined by straight line segments. Each segment (bounded by two data points) can be interpolated independently. The parameter *mu* defines where to estimate the value on the interpolated line, it is 0 at the first point and 1 and the second point. For interpolated values between the two points *mu* ranges between 0 and 1.

$$y(x) = y_1 \times (1 - mu) + y_2 \times mu \tag{10.1}$$

2. **Nearest Neighbors**
   The nearest neighbors (NN) method predicts the value of an attribute at an arbitrary point based on the value of the nearest sample by drawing perpendicular bisectors between sampled points (n), forming such as Thiessen (or Dirichlet/Voronoi) polygons ($V_i, i = 1, 2, \ldots, n$). This produces one polygon per sample and the sample is located in the center of the polygon, such that in each polygon all points are nearer to its enclosed sample point than to any other sample points [14–16]. The estimations of the attribute at arbitrary points within polygon V_i are the measured value at the nearest single sampled data point x_i that is $\hat{z}(x_0) = z(x_i)$. The weights are:

$$\lambda_i = \begin{cases} 1 & \text{if } x_i \in V_i \\ 0 & \text{otherwise} \end{cases} \tag{10.2}$$

   All points (or locations) within each polygon are assigned the same value [15, 16]. A number of algorithms exist to generate the polygons [17], including pycnophylactic interpolation [18].
3. **Natural**
   The natural neighbors method was introduced by Sibson (1981). It combines the best features of NN and Triangular Irregular Network [16]. The first step is a triangulation of the data by Delauneys method, in which the apices of the triangles are the sample points in adjacent Thiessen polygons. This triangulation is unique except where the data are on a regular rectangular grid. To estimate the value of a point, it is inserted into the tessellation and then its value is determined by sample points within its bounding polygons. For each neighbors, the area of the portion of its original polygon that became incorporated in the tile of the

new point is calculated. These areas are scaled to sum 1 and are used as weights for the corresponding samples [16]. This method can provide a more smooth approximation to the underlying "true" function.

4. **Cubic**
   A cubic spline is a spline constructed of piecewise third-order polynomials which pass through a set of N control points. The polynomials describe pieces of a line or surface (i.e., they are fitted to a small number of data points exactly) and are fitted together so that they join smoothly [16, 18]. The places where the pieces join are called knots. The choice of knots is arbitrary and may have a dramatic impact on the estimation [18]. Splines with few knots are generally smoother than splines with many knots; however, increasing the number of knots usually increases the fit of the spline function to the data. Knots give the curve freedom to bend to more closely follow the data.

In this work we propose a Cellular Automata (CA) based method for shaping accurate morphology maps using scattered data collected from multi robot teams. CA have attracted researchers from several disciplines (e.g., from the field of robotics [19, 20], image processing [21, 22] and environmental modelling [23]) and a large number of scientific papers are published every year.

CA, initially were proposed as models of physical systems, where space and time are discrete and interactions are local, by von Neumann [24]. Any physical system satisfying differential equations may be approximated by a CA, by introducing finite differences and discrete variables [25–31]. Additionally, CA are one of the computational structures best suited for a VLSI realization [32–35]. The CA architecture offers a number of advantages and beneficial features such as simplicity, regularity, ease of mask generation, silicon-area utilization, and locality of interconnections [26, 33].

In order to evaluate the proposed approach, experiments conducted employing real-word data collected from two different types of robot teams. Initially, using the proposed CA based method for shaping accurate morphology maps, a detailed map of a large unknown underwater area in Oporto, Portugal was constructed. In the sequel, the proposed approach generates a detailed map using data collected from an area near Zurich, Switzerland, by a team of aerial robots. It is worth noting that the collected data, in the two experimental setups, are captured employing different type of sensors. In both cases, the proposed CA based method outperforms the performance of several other well-known methods from the literature.

The rest of the chapter is organized as follows. Section 10.2.1 demonstrates the problem of scattered data interpolation in strict notation. In Sect. 10.2.2, we demonstrate the exact steps of the proposed methodology using CA. Section 10.3 presents a series of experiments carried out with measurements from real robot systems. Conclusions and future steps are given in Sect. 10.4.

10.2 CA Based Methodology for Shaping Morphology Maps Using Scattered Data

10.2.1 Problem Formulation

Without loss of generality, we can assume that the area to be mapped is constrained within a rectangle in the (x, y)-coordinates, i.e., the mobile robots are called to map the area constrained in the (x, y)-coordinates as follows:

$$\mathcal{U} = \left\{ x, y : x \in [x_{min}, x_{max}], y \in [y_{min}, y_{max}] \right\} \tag{10.3}$$

This rectangle can be divided into discrete cells, in such a way that if all the values in each cell are known, the representation of this rectangle would approximate the real surface.

$$\begin{aligned} x_{i+1} &= x_i + \Delta x, \Delta x = \frac{(x_{max} - x_{min})}{L} \\ y_{i+1} &= y_i + \Delta y, \Delta y = \frac{(y_{max} - y_{min})}{M} \end{aligned} \tag{10.4}$$

where L and M denote the desirable discretization in x and y axis correspondingly. The goal is from an, arbitrarily located, set of data (x_i, y_i, f_i) , $i = 1, \ldots, N$ that represent the error-free[1] measurements taken from different type of sensors, to generate the values $(x, y) \in \mathcal{U}$ in all the cells of the rectangle.

10.2.2 Proposed Methodology

The basic steps of the proposed CA methodology are given as follows:

STEP 1: The map area is divided into a matrix, and for now on it will be referred as $\mathcal{C}$, of identical square cells that represented by a CA, where each cell of the map is considered as a CA cell.

STEP 2: In the second step is applied the registration between the measurements' data and the corresponding CA cells. In other words every set (x_i, y_i, f_i) of measurements has to be placed in the appropriate cell $\mathcal{C}\,(x_i, y_i) = f_i$. After this step we have defined the dimensions of our CA and its initial conditions.

STEP 3: The evolution rules of a m CA cell, where $m : (m_x, m_y)$, are chosen as a combination of two different approaches. On the one hand, a direct "propagation" of the information is applied, following the Moore's neighbor (Eq. 10.5), around the initially known cells. Please note that the logical expression: $||i - m_x|| = 1$ or

[1] We will assume that the robot's measurements are filtered and free of bias/Gaussian noise. It has to be emphasized that the proposed approach can be extended to deal with noisy data giving weights about the confidence level of the measurement's accuracy.

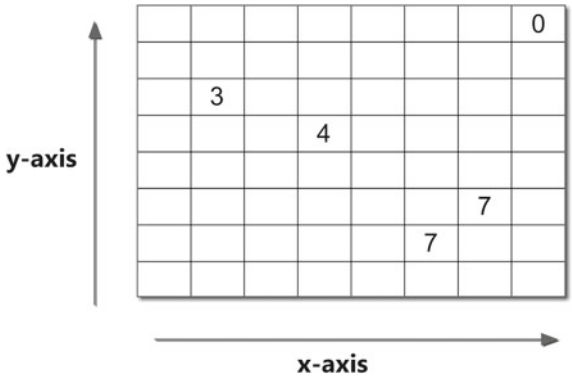

Fig. 10.2 The initial conditions of CA

$||j - m_y|| = 1$ has to be true in order to follow the Moore's neighborhood. If a cell is affected by more than one value in the same step, a simple average is applied (see Fig. 10.3).

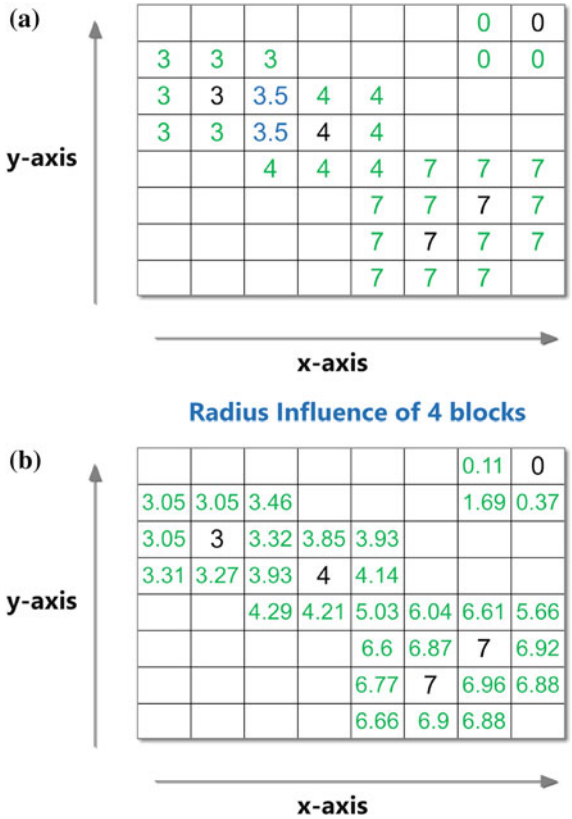

Fig. 10.3 The evolution rules. **a** Direct "propagation". **b** Using Remote Information

$$\mathcal{C}_1^{t_i}(m_x, m_y) = \begin{cases} \mathcal{C}^{t_i}(i, j) & \text{if } \mathcal{C}_1^{t_i-1}(m_x, m_y) = 0 \\ \mathcal{C}_1^{t_i-1}(m_x, m_y) & \text{otherwise} \end{cases} \tag{10.5}$$

On the other hand, an evolution rule using information of the closest neighbors is applied. We calculate the estimated value using a weighted average of the neighbors that are in a pre-specified (see 10.2.3) "radius of influence" (*RD*) as shown in Eq. 10.6:

$$\mathcal{C}_2^{t_i}(m_x, m_y) = \frac{\sum_{k=1}^{n} \mathcal{C}^{t_{i-1}}\left(k_x, k_y\right) / M_d(k, m)^2}{\sum_{k=1}^{n} 1/M_d(k, m)^2} \tag{10.6}$$

STEP 4: Subsequently, a merging procedure is applied in order to render the final value of the estimated cell, which, can be represented as follows:

$$\mathcal{C}^{t_i}(m_x, m_y) = a \times \mathcal{C}_1^{t_i}(m_x, m_y) + (1 - a) \times \mathcal{C}_2^{t_i}(m_x, m_y) \tag{10.7}$$

Here, a servers as *smoothing factor* to give more/less weight to the one term against the other.

STEP 5: Repeat *STEP 3–4* until every cell of the CA obtain an estimation about its height.

10.2.3 Define Adaptively the "Radius of Influence"

The *RD* corresponds to the maximum distance that is allowed between the current CA cell and every other cell with value already calculated. It defines how far "travels" the information, from the measurements, on the terrain. There is no global value for the *RD* that can be applied in every map, e.g., if the area to be mapped was "flat", probably a good strategy would be to choose a "big" value for the *RD*. Information about the morphology of the area, and thus about the *RD*, can be derived by exploiting the distribution of the measurements' set. The following steps are describing the dynamic adjustment of the *RD*, utilizing the robot's measurements.

STEP 1: The measurements sets have specific attributes, that could differ from a sub-area to another. Initially these measurements data have to be classified, in an optimal manner. To keep the analysis as general as possible, it will be considered that the number of classes is unknown, and have to be investigated as the initial measurements change. Based on the above we can formulate the following optimization problem,

$$\begin{aligned} &\underset{k}{\text{minimize}}\ \mathcal{F}(k) = \textstyle\sum_{i=1}^{N} ||\,[x_c\ y_c\ h_c]^i - [x_i\ y_i\ h_i]\,|| \\ &\text{subject to } k \geq 2 \end{aligned} \tag{10.8}$$

where N is the number of the measurements, k denotes the number of centroids, $[x_i\ y_i\ h_i]$ denotes the i-th measurement vector and $[x_c\ y_c\ h_c]^i$ the centroid of the

class, where with the current centroids' selection, belongs the i-th measurement. The cost function $\mathcal{F}(k)$ can be separated in two terms.

Whenever an updated value, about the number of classes, is calculated, the algorithm K-means is called to find the 3 dimension vector of each centroid. Only when the iterative procedure of K-means ends, the cost function $\mathcal{F}(k)$ (Eq. 10.8) is calculated again to evaluate the difference in classification with the modified number of centroids.

Taking into account that the dimensions of the centroids are in three dimensional space, the above procedure can be completed in a reasonable time using an optimization algorithm such as Hill climbing.

STEP 2: Having defined the optimal number of classes k_{opt} and the corresponding centroids (using K-means) we can now proceed to the final calculation about the *RD*.

$$RD = \frac{\mathcal{F}(k_{opt})}{N} \tag{10.9}$$

The *RD* corresponds to the influence of the current cell in its neighborhood. If we calculate the average of distances between all the measurements and their centroids (Eq. 10.9), we can have a rough estimation about the spatial influence of every cell around its neighborhood.

10.3 Experiments

In this section we validate the proposed approach using real data collected by two different types of robotic devices using different sensors. In both cases the robots were collecting the data in order to construct a map to assist them in performing a predefined mission. The first case refers to data collected from Oporto's harbor area using bathymetry sensors [36], while the second test case refers to data collected using a camera mounted on a single aerial robot [11]. The diversity of the data collected from two different type of devices is used as a proof of concept of the generality and applicability of our approach.

10.3.1 Underwater Scenario—Oporto harbor

Using the method described in detail in Sect. 10.2.2, we have reconstructed the undersea morphology of a sub-region of Oporto's harbor. The underwater region covers an area of 200 × 200 points spaced by 5 m. To acquire full knowledge of the seafloor we should have known the value in each of the 40,000 points arising from the previous grid. In Fig. 10.4 we present the real morphology of the area and the information which will be used as ground truth to evaluate our approach. In a realistic scenario the robot team, in this case a team of autonomous underwater vehicles, would have returned after the completion of a mission with a data set much smaller

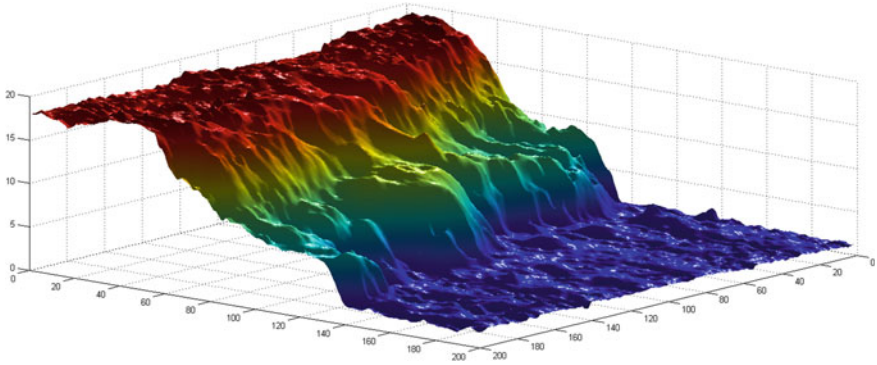

Fig. 10.4 Ground truth—40,000 points—real representation of the operation area

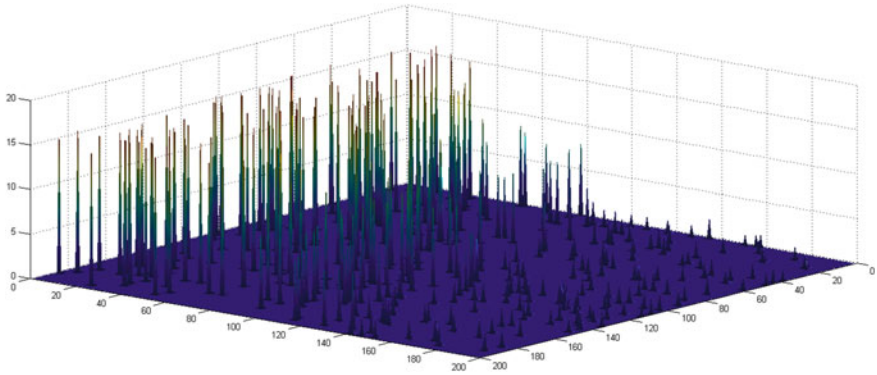

Fig. 10.5 Initial measurements—308 points gathered by swarm of robots

than the one needed to have a detailed representation of the environment monitored. Figure 10.5 depicts the visualization of real scenario where the AUVs gathered 308 measurements points. It is worth-noticing that this subset (derived from robot's measurements) constitutes only the 0.77 % [2] of the total area to be mapped, and for that is considered a severe experiment.

The details of the CA environment are as follows:

- 2-D CA with 200 × 200 cells
- The initial condition of the CA are the 308 measurements as depicted in Fig. 10.5 and the unknown cells are filled with −1
- The evolution rules are as explained in the previous section (Eq. 10.7)

[2] Note that now and in the next experiments there has not been any analysis about the distribution that is followed by the measurements. Different modalities, like different number of robots or different type of sensors, etc., will lead to a different data distribution. The above problem is tackled by conducted the same experiment 500 times with the initial measurements stochastically changed and keep the average of error.

The Fig. 10.6 demonstrates the incremental evolution of the CA over the time. The whole procedure, even in this case with the extremely sparse measurements data, is completed in 20 time-steps, renders the procedure directly applicable to real-time interpolation systems.

As evaluation, the Fig. 10.7 illustrates the results of different interpolation methods for the same data sets. The visual superiority of the proposed method is appeared also in extended simulations, where a batch of 500 iterations, with different initial measurements (conditions), per method is conducted and the average of L^2 error with ground-truth is calculated. The results are shown in Table 10.1.

10.3.2 Aerial robots Scenario

To test the efficiency and stability of the proposed approach, regardless of the area which is called to reconstruct the morphology, we have also tested it in data gathered from aerial robots. In this case we will demonstrate the reconstruction of a "village like" area based on data collected in an area near Zurich, Switzerland. The initial data and the respective map were collected using a state-of-the-art visual-SLAM algorithm which tracks the pose of the camera while simultaneously and autonomously, building an incremental map of the surrounding environment. More details regarding the extraction methodology are given in [8, 37].

This area is consisted of ruins and small urban structures and for that has high rate of inhomogeneity and the transitions among different sub-areas are often steep. For those reasons the interpolation in measurements' data consists in a very challenging task. In Fig. 10.8 is presented the real surface with 13,855 points Fig. 10.8a and the reconstructed one from 462 (3.3 %) measurements: using the proposed method Fig. 10.8b, while in Fig. 10.9 as evaluation we illustrate the results of the different interpolation methods for the same data sets. The average of L^2-*Norm* for this initial configuration for each method is presented in the second row of Table 10.2. The Table 10.2 is an analysis of the impact that has the reduction of the initial measurements in the final result of the interpolation procedure.

As robots are reducing their sampling rates of their sensors or operating for less time the pool of gathered measurements will be smaller and thus the interpolation procedure will have less accuracy, regardless of the choice of the interpolation method and that's something that is imprinted in the Table 10.2. The proposed method achieves better performance not only as weighted average, but in every different configuration in number of measurements. This property can be achieved because the proposed method can adapt its parameters in the available data set (Sect. 10.2.3) and finally is able to manage to elaborate them in a efficient fashion.

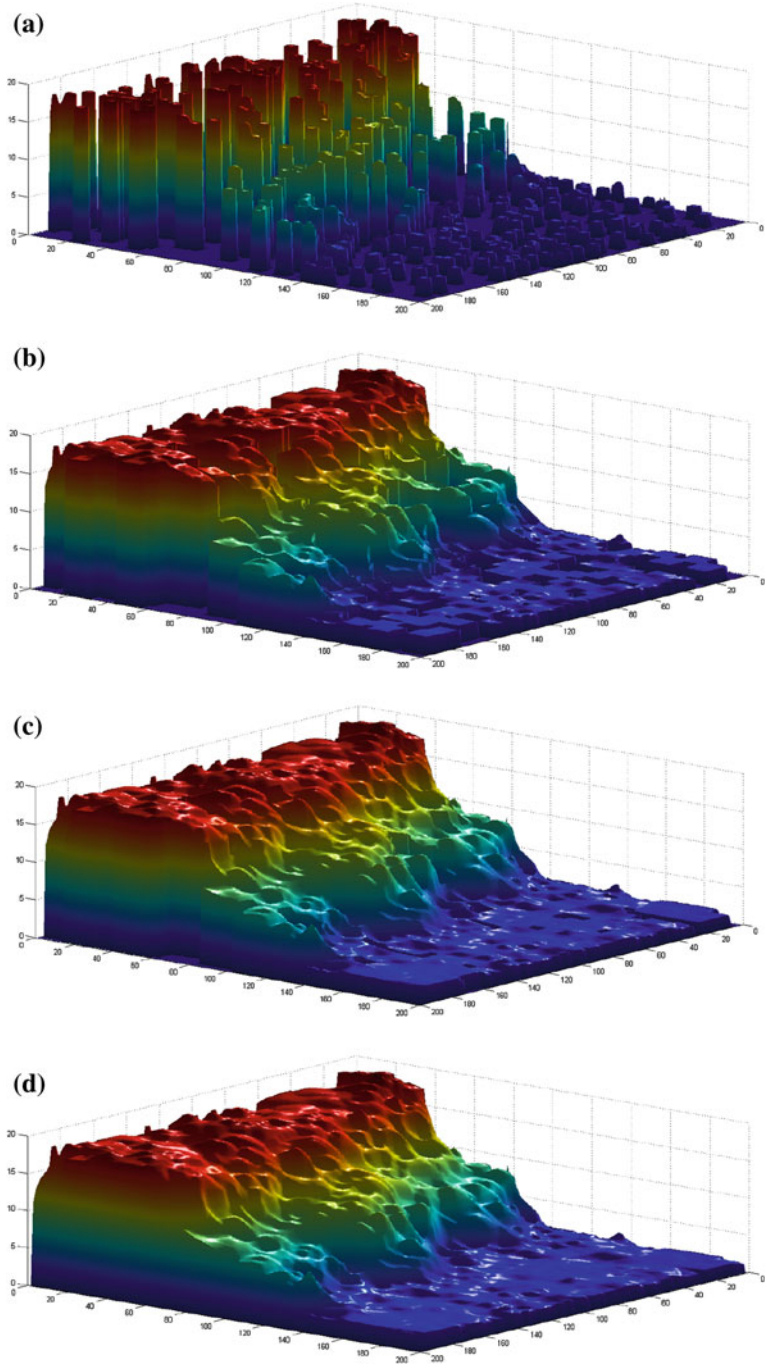

Fig. 10.6 Incremental evolution based on the CA methodology. The figures illustrate the progress at 25, 50, 75 and 100 %, correspondingly, of the CA process

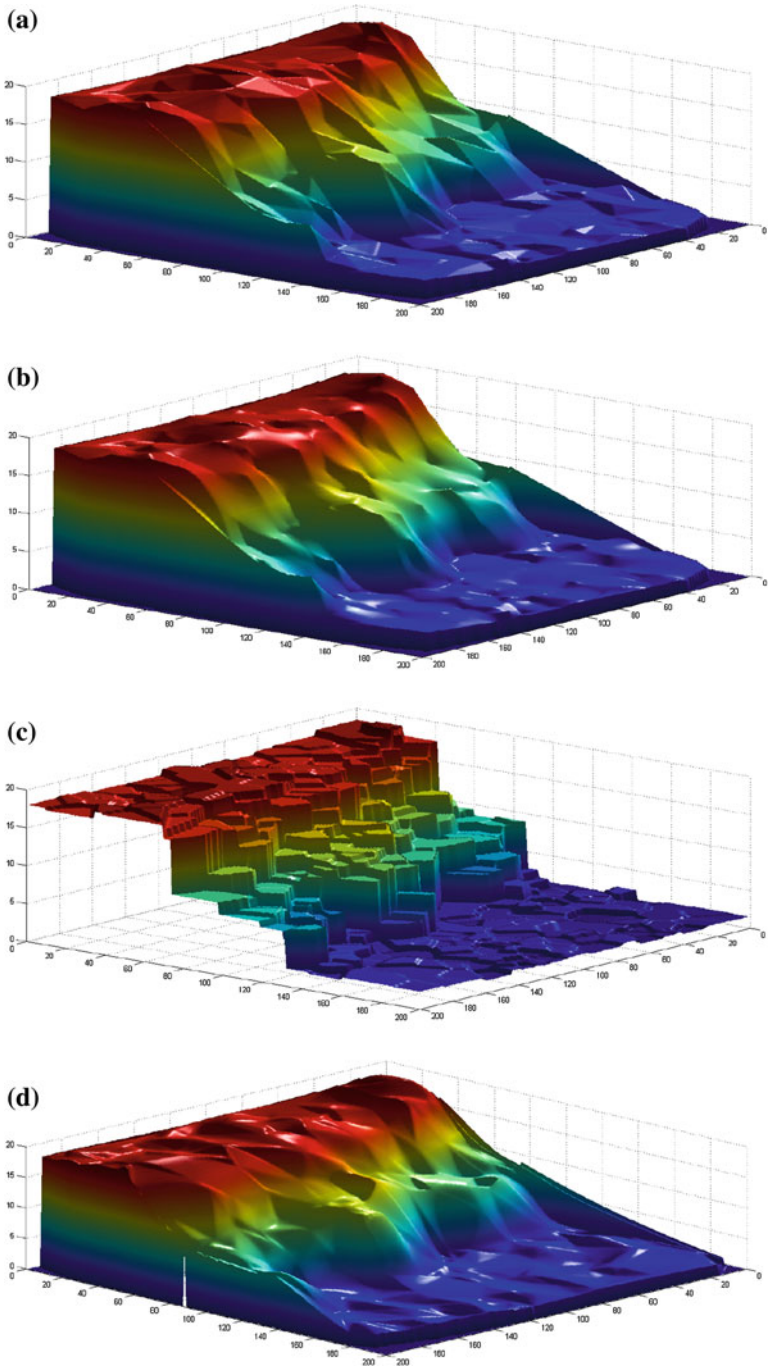

Fig. 10.7 Results using other interpolation methods. **a** Linear, **b** Natural, **c** Nearest neighbors, **d** Cubic

Table 10.1 L^2-*Norm* between the ground truth and the constructed map for each method for each number of initial measurements

L^2-*Norm*	Linear	Natural	Nearest neighbors	Cubic	Proposed
Weighted average	340.1553	340.8651	288.520	338.520	266.9886

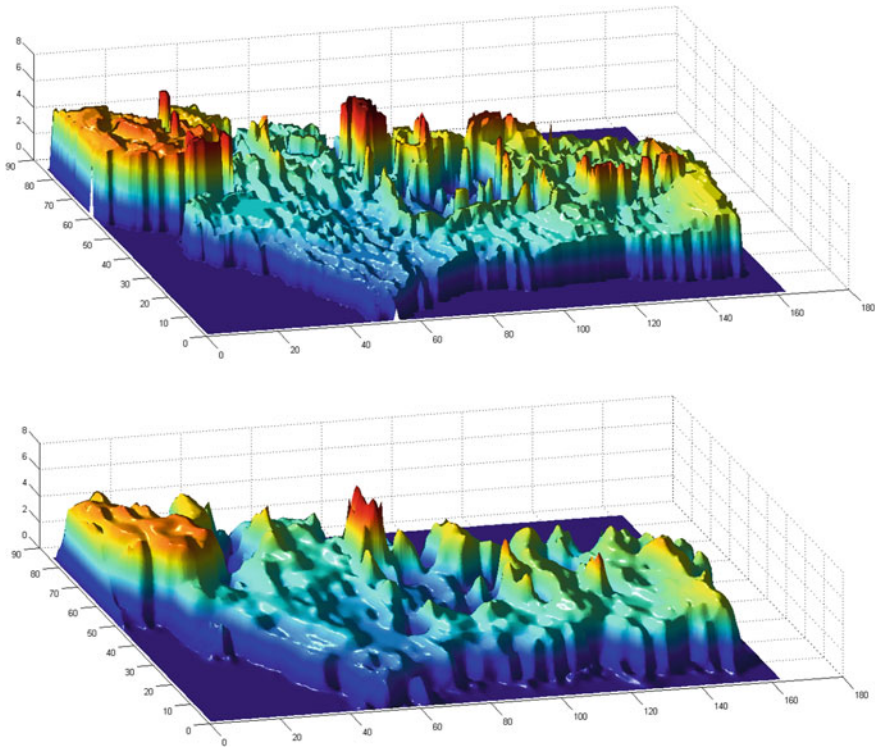

Fig. 10.8 Zurich area—interpolation results over 462 points (3.3 %), **a** Ground truth **b** Proposed methodology using CA

10.4 Conclusions and Future Work

In this chapter, a novel method is presented using CA for scattered data interpolation. We have successfully demonstrated the composition of morphology maps from sensor's measurements that outperform the most common used ones in all the test-cases. The efficiency of the methodology relies, both on the ability of CA to efficiently process elements that are arranged in a regular grid of identical cells, and on the adaptability on the local morphology of each region, analyzing the variety in measurements.

We are interested in considering situations where the robots' measurements contain errors. In this case, the problem becomes even harder since the CA now have

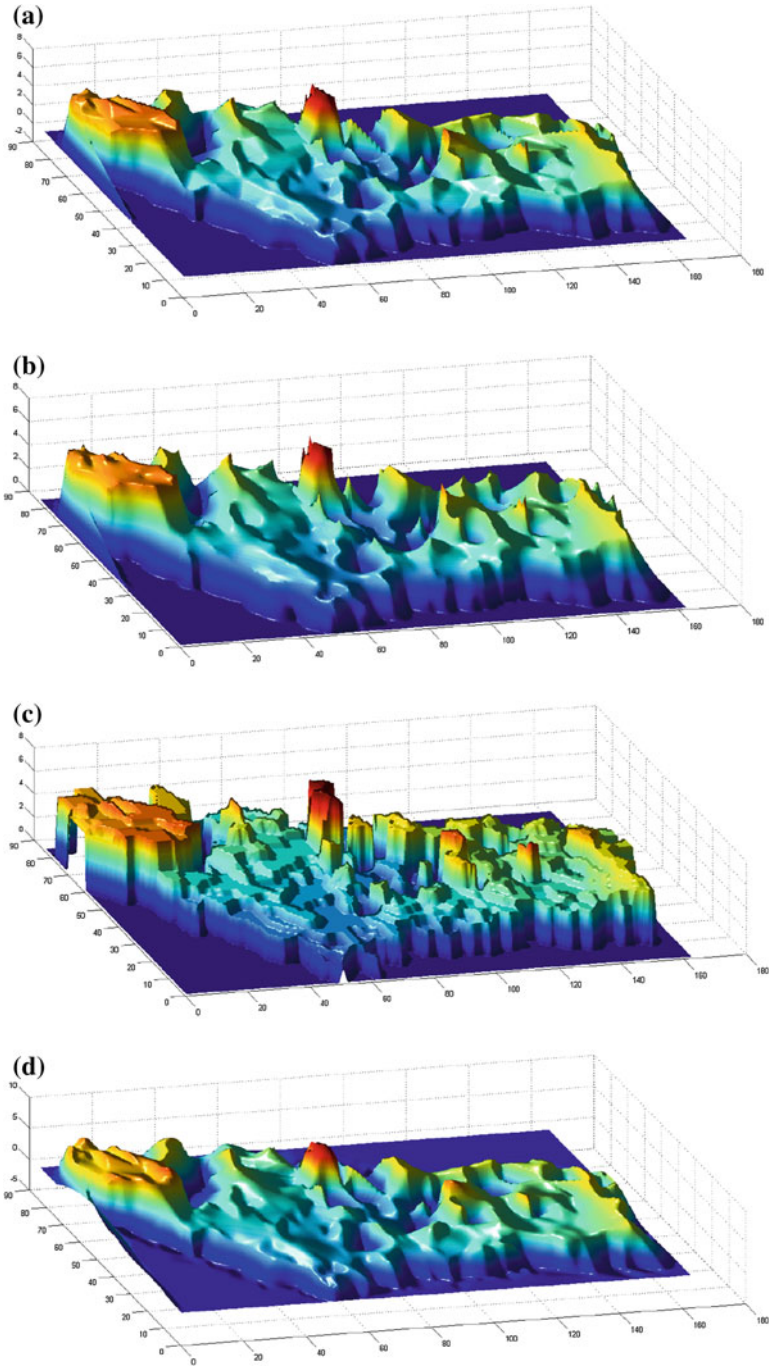

Fig. 10.9 Zurich area—Results using other interpolation methods, namely **a** Linear, **b** Natural, **c** Nearest neighbors, **d** Cubic

Table 10.2 L^2-*Norm* between the ground truth and the constructed map for each method for each number of initial measurements

Initial points %	Linear	Natural	Nearest neighbors	Cubic	Proposed
1386 (10)	361.117	343.837	407.426	355.361	338.004
462 (3.3)	499.623	477.9376	562.482	507.109	463.655
277 (2)	593.827	571.870	656.690	602.876	544.507
139 (1)	754.819	734.043	826.645	772.014	699.616
69 (0.5)	959.421	938.274	1032.046	933.780	866.569
46 (0.33)	1108.377	1086.865	1140.229	1079.947	928.137
Weighted average	*720.022*	*699.247*	*778.214*	*715.578*	*657.203*

to solve two problems. In the case CA will have to face a dual problem, since they will have to identify which measurements are useful in the process and which have to be ignored or corrected and also must be able to make an estimation, followed the proposed methodology, about the morphology of the terrain, in real-time to keep its directly applicable nature. We would like also to investigate scenarios in which the objective is to build the morphology map of sub-region, where the robots do not visit at all (extrapolation) or in which the environment changes over time.

References

1. Gasca, M., Sauer, T.: Polynomial interpolation in several variables. Adv. Comput. Math. **12**(4), 377–410 (2000)
2. Lehmann, T.M., Gonner, C., Spitzer, K.: Survey: Interpolation methods in medical image processing. Med. Imaging IEEE Trans. **18**(11), 1049–1075 (1999)
3. Franke, R., Nielson, G.M.: Scattered data interpolation and applications: a tutorial and survey. In: Geometric Modeling, pp. 131–160. Springer, Berlin (1991)
4. Thorsten, L., Michael, H., Wuensche, H.-J.: Autonomous ground vehicles concepts and a path to the future. Proc. IEEE **100**(13), 1831–1839 (2012)
5. Achtelik, M., Achtelik, M., Brunet, Y., Chli, M., Chatzichristofis, S.A.., Decotignie, J-D., Doth, K-M., Fraundorfer, F., Kneip, L., Gurdan, D., Heng, L., Kosmatopoulos, E.B., Doitsidis, L., Lee, G.H., Lynen, S., Martinelli, A., Meier, L., Pollefeys, M., Piguet, D., Renzaglia, A., Scaramuzza, D., Siegwart, R., Stumpf, J., Tanskanen, P., Troiani, C., Weiss. S.: Sfly: swarm of micro flying robots. In: IROS, pp. 2649–2650. IEEE (2012)
6. Birk, A., Pfingsthorn, M., Bülow, H.: Advances in underwater mapping and their application potential for safety, security, and rescue robotics. In: IEEE International Symposium on Safety, Security, Rescue Robotics (SSRR). IEEE Press (2012)
7. Michael, N., Shaojie, S., Mohta, K., Mulgaonkar, Y., Kumar, V., Nagatani, K., Okada, Y., Kiribayashi, S., Otake, K., Yoshida, K., Ohno, K., Takeuchi, E., Tadokoro, S.: Collaborative mapping of an earthquake-damaged building via ground and aerial robots. J. Field Robot. **29**(5), 832–841 (2012)
8. Blosch, M., Weiss, S., Scaramuzza, D., Siegwart, R.: Vision based MAV navigation in unknown and unstructured environments. In: IEEE International Conference on Robotics and Automation (ICRA), 2010, pp. 21–28. IEEE (2010)

9. Fraundorfer, F., Heng, L., Honegger, D., Lee, G.H., Meier, L., Tanskanen, P., Pollefeys, M.: Vision-based autonomous mapping and exploration using a quadrotor mav. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012, pp. 4557–4564. IEEE (2012)
10. Majdik, A., Albers-Schoenberg, Y., Scaramuzza, D.: MAV urban localization from google street view data. In: IROS, pp. 3979–3986 (2013)
11. Doitsidis, L., Weiss, S., Renzaglia, A., Achtelik, M.W., Kosmatopoulos, E.B., Siegwart, R., Scaramuzza, D.: Optimal surveillance coverage for teams of micro aerial vehicles in GPS-denied environments using onboard vision. Auton. Robots **33**(1–2), 173–178 (2012)
12. Akyildiz, I.F., Pompili, D., Melodia, T.: Underwater acoustic sensor networks: research challenges. Adhoc Netw. **3**(3), 257–279 (2005)
13. Kapoutsis, A.Ch., Chatzichristofis, S.A., Doitsidis, L., Borges de Sousa, J., Kosmatopoulos, E.B.: Autonomous navigation of teams of unmanned aerial or underwater vehicles for exploration of unknown static & dynamic environments. In: 21st Mediterranean Conference on Control & Automation (MED), 2013, pp. 1181–1188. IEEE (2013)
14. Bohling, G.: Introduction to Geostatistics and Variogram Analysis, p. 20. Kansas Geological Survey, Kansas (2005)
15. Ripley, B.D.: Spatial Statistics, vol. 575. Wiley.com, New York (2005)
16. Webster, R., Oliver, M.A.: Geostatistics for Environmental Scientists. Wiley, Chichester (2007)
17. Christopher, C.M., Condal, A.R.: A spatial data structure integrating GIS and simulation in a marine environment. Mar. Geodesy **18**(3), 213–228 (1995)
18. Burrough, P.A.: Principles of Geographical Information Systems for Land Resources Assessment (1986)
19. Charalampous, K., Amanatiadis, A., Gasteratos, A.: Efficient robot path planning in the presence of dynamically expanding obstacles. In: ACRI, volume 7495 of Lecture Notes in Computer Science, pp. 330–339. Springer (2012)
20. Ioannidis, K., Sirakoulis, GCh., Andreadis, I.: Cellular automata-based architecture for cooperative miniature robots. J. Cell. Autom. **8**(1–2), 91–111 (2013)
21. Chatzichristofis, S.A., Mitzias, D.A., Sirakoulis, GCh., Boutalis, Y.S.: A novel cellular automata based technique for visual multimedia content encryption. Opt. Commun. **283**(21), 4250–4260 (2010)
22. Zagoris, K., Pratikakis, I.: Scene text detection on images using cellular automata. In: ACRI, pp. 514–523 (2012)
23. Georgoudas, I.G., Sirakoulis, GCh., Scordilis, E.M., Andreadis, I.: A cellular automaton simulation tool for modelling seismicity in the region of Xanthi. Environ. Modell. Softw. **22**(10), 1455–1464 (2007)
24. Von Neumann, J., Burks, A.W., et al.: Theory of Self-Reproducing Automata. University of Illinois Press, Urbana (1966)
25. Chopard, B., Droz, M.: Cellular Automata Modeling of Physical Systems. Cambridge University Press, New York (1998)
26. Toffoli, T.: Cellular automata as an alternative to (rather than an approximation of) differential equations in modeling physics. Phys. D Nonlinear Phenom. **10**(1–2), 117–127 (1984)
27. Bialynicki-Birula, I.: Weyl, Dirac, and Maxwell equations on a lattice as unitary cellular automata. Phys. Rev. D **49**(12), 6920–6927 (1994)
28. Omohundro, S.: Modelling cellular automata with partial differential equations. Phys. D Nonlinear Phenom. **10**(1–2), 128–134 (1984)
29. Malamud, B.D., Turcotte, D.L.: Cellular-automata models applied to natural hazards. Comput. Sci. Eng. **2**(3), 42–51 (2000)
30. Sirakoulis, GCh., Karafyllidis, I., Mardiris, V., Thanailakis, A.: Study of lithography profiles developed on non-planar Si surfaces. Nanotechnology **10**, 421–427 (1999)
31. Sirakoulis, GCh., Karafyllidis, I., Thanailakis, A.: A cellular automaton model for the effects of population movement and vaccination on epidemic propagation. Ecol. Modell. **133**(3), 209–223 (2000)

32. Sirakoulis, GCh., Karafyllidis, I., Thanailakis, A., Mardiris, V.: A methodology for VLSI implementation of cellular automata algorithms using VHDL. Adv. Eng. Softw. **32**(3), 189–202 (2000)
33. Sirakoulis, GCh., Karafyllidis, I., Thanailakis, A.: A CAD system for the construction and VLSI implementation of cellular automata algorithms using VHDL. Microprocess. Microsyst. **27**(8), 381–396 (2003)
34. Sirakoulis, GCh.: A TCAD system for VLSI implementation of the CVD process using VHDL. Integr. VLSI J. **37**(1), 63–81 (2004)
35. Mardiris, V., Sirakoulis, GCh., Mizas, Ch., Karafyllidis, I., Thanailakis, A.: A CAD system for modeling and simualtion of computer networks using cellular automata. IEEE Trans. SMC-Part C **38**(2), 1–12 (2008)
36. Glynn, J., de Moustier, C., Huff, L.: Survey operations and results using a Klein 5410 bathymetric sidescan sonar. In: US Hydro (2007)
37. Weiss, S., Achtelik, M., Kneip, L., Scaramuzza, D., Siegwart, R.: Intuitive 3D maps for MAV terrain exploration and obstacle avoidance. J. Intell. Robot. Syst. **61**(1–4), 473–493 (2011)