A System Architecture for the Monitoring of Continuous Phenomena by Sensor Data Streams

Dem Fachbereich Mathematik/Informatik der
Universität Osnabrück
zur Erlangung des akademischen Grades eines
Dr. rer. nat.

eingereichte Dissertation

von

Herrn Peter Lorkowski M.Sc.

Datum der Einreichung: 23.05.2018

Referent: Prof. Dr.-Ing. Manfred Ehlers (em.)
Koreferent: Prof. Dr. Thomas Brinkhoff

Tag der mündlichen Prüfung: 14.12.2018

Für meine Großmutter Franziska

Danksagungen

Mein erster Dank gebührt Thomas Brinkhoff, der mit dem Vorschlag zur Promotion sein Vertrauen in meine Fähigkeiten zeigte, welches mich im Verlauf der Arbeit stets getragen und motiviert hat. Er fand die Balance zwischen Freiheit und Ansporn und war ein inspirierter und inspirierender Gesprächspartner. Manfred Ehlers hat mich durch seine aufgeschlossene und konstruktive Art dazu ermutigt, für meine Arbeit eine weitere Perspektive zu suchen. Die dadurch angeregte Recherche auch jenseits des recht engen Fachzweiges hat mich sowohl fachlich als auch persönlich sehr bereichert. Thomas Jarmer möchte ich für die wertvollen Ratschläge im Rahmen des Doktorandenkolloquiums danken. Die weiteren Experimente, die ich daraufhin durchführte, haben zu wichtigen neuen Erkenntnissen geführt und die Arbeit um einen wesentlichen Aspekt ergänzt. Im Rahmen einer Lehrveranstaltung von Ute Vogel lernte ich die elegante Methodik der Geostatistik kennen, die mich von da an in meiner fachlichen Entwicklung begleitet hat. Ihre ansteckende Begeisterung und motivierende Art der Vermittlung prägten mein Interesse an dem Thema nachhaltig. Helge Borman hatte stets ein offenes Ohr für meine Ideen und konnte ihnen mit konstruktiver Kritik so manches Mal auf die Sprünge helfen. Katharina Henneböhl verdanke ich wertvolle Hinweise bezüglich geostatistischer Verfahren ebenso wie so manche moralische Unterstützung. Andreas Gollenstede danke ich für die zahlreichen inspirierenden fachlichen und philosophischen Exkurse, die sowohl apollinische als auch dionysische Züge trugen. Mit Manuel Sellmann verbrachte ich viele Stunden anregender Diskussionen über fachliche und organisatorische Belange, aber auch einige Zeit bei zuweilen dringend notwendiger musikalischer Zerstreuung. Zum anregenden und oft genug befreienden Austausch über Fluch und Segen von Freiheit und Komplexität von Informationssystemen war Tobias Werner mir stets ein wertvoller Gesprächspartner und Ratgeber. Meiner Frau Christina danke ich für Geduld, Zuversicht und Zuspruch besonders in den schwierigen Phasen der Arbeit, so dass ich immer wieder neue Motivation finden konnte. Zur allgemeinen Erheiterung trug stets unsere kleine Tochter Elisa bei. Ohne eine gewisse Hartnäckigkeit und Zielstrebigkeit, die ich meinen Eltern zu verdanken habe, hätte ich diese Aufgabe sicherlich nicht bewältigen können.

The game of science is, in principle, without end. He who decides one day that scientific statements do not call for any further test, and that they can be regarded as finally verified, retires from the game.

(Karl Popper, The Logic of Scientific Discovery)

Abstract

The monitoring of continuous phenomena like temperature, air pollution, precipitation, soil moisture etc. is of growing importance. Decreasing costs for sensors and associated infrastructure increase the availability of observational data. These data can only rarely be used directly for analysis, but need to be interpolated to cover a region in space and/or time without gaps. So the objective of monitoring in a broader sense is to provide data about the observed phenomenon in such an enhanced form.

Notwithstanding the improvements in information and communication technology, monitoring always has to function under limited resources, namely: number of sensors, number of observations, computational capacity, time, data bandwidth, and storage space. To best exploit those limited resources, a monitoring system needs to strive for efficiency concerning sampling, hardware, algorithms, parameters, and storage formats.

In that regard, this work proposes and evaluates solutions for several problems associated with the monitoring of continuous phenomena. Synthetic random fields can serve as reference models on which monitoring can be simulated and exactly evaluated. For this purpose, a generator is introduced that can create such fields with arbitrary dynamism and resolution. For efficient sampling, an estimator for the minimum density of observations is derived from the extension and dynamism of the observed field. In order to adapt the interpolation to the given observations, a generic algorithm for the fitting of kriging parameters is set out. A sequential model merging algorithm based on the kriging variance is introduced to mitigate big workloads and also to support subsequent and seamless updates of real-time models by new observations. For efficient storage utilization, a compression method is suggested. It is designed for the specific structure of field observations and supports progressive decompression.

The unlimited diversity of possible configurations of the features above calls for an integrated approach for systematic variation and evaluation. A generic tool for organizing and manipulating configurational elements in arbitrary complex hierarchical structures is proposed. Beside the root mean square error

(RMSE) as crucial quality indicator, also the computational workload is quantified in a manner that allows an analytical estimation of execution time for different parallel environments.

In summary, a powerful framework for the monitoring of continuous phenomena is outlined. With its tools for systematic variation and evaluation it supports continuous efficiency improvement.

environmental monitoring, spatio-temporal interpolation, sensor data stream, kriging variance, computational efficiency

Zusammenfassung

Das Monitoring kontinuierlicher Phänomene wie Temperatur, Verteilung von Luftschadstoffen, Niederschlag, Bodenfeuchte etc. gewinnt zunehmend an Bedeutung. Bei sinkenden Kosten für Sensoren und Kommunikationsinfrastruktur nimmt die Verfügbarkeit von entsprechenden Messdaten stetig zu. Eine unmittelbare Nutzung dieser Messdaten ist jedoch nur selten möglich; für viele Analysen müssen sie interpoliert werden, um einen Bereich räumlich und/ oder zeitlich lückenlos abzudecken. So besteht die Aufgabe eines Monitorings im weiteren Sinne auch darin, die beobachtete Variable in einer solchen lückenlosen Form bereitzustellen.

Trotz stetigem Fortschritt der Informations- und Kommunikationstechnologie bleibt ein Monitoring stets begrenzten Ressourcen unterworfen: Anzahl der Sensoren und Beobachtungen, Rechenleistung, Zeit, Datenraten und Speicherplatz. Für eine bestmögliche Nutzung der jeweils verfügbaren Ressourcen sollte stets eine hohe Effizienz bezüglich der Sensorik, der Hardware, der Algorithmen und zugehöriger Parameter sowie der Speicherformate angestrebt werden.

In Bezug auf diese Problemstellung werden verschiedene Lösungsansätze erarbeitet und evaluiert. Synthetische kontinuierliche Zufallsfelder dienen dabei als Referenz, um die Qualität und Effizienz des darauf simulierten Monitorings exakt quantifizieren zu können. Es wird ein Generator vorgestellt, der Zufallsfelder beliebiger Dynamik und Auflösung erzeugt. Für eine möglichst effiziente Messanordnung wird ein Schätzer für die minimale Beobachtungsdichte aus der Ausdehnung und Dynamik des beobachteten Feldes abgeleitet. Für eine gute Adaption der Interpolation an die durch die Beobachtungen gegebenen statistischen Eigenschaften wird ein generischer Algorithmus zur Parameterschätzung des Kriging-Interpolators vorgestellt. Ein sequentieller Algorithmus zur Verschmelzung mehrerer Interpolationsergebnisse eines Bereichs kann den Berechnungsaufwand reduzieren und kann außerdem verwendet werden, um in einem Datenstromsystem kontinuierlich und nahtlos neue Beobachtungen in ein Echtzeit-Modell zu integrieren. Zur effizienteren Nutzung von Speicherplatz wurde ein Kompressionsverfahren entwickelt. Es nutzt die spezifischen Eigenschaften der Beobachtungsdaten von kontinuierlichen Phänomenen und unterstützt eine progressive Dekompression.

Die erwähnten Werkzeuge bieten prinzipiell eine unbegrenzte Vielfalt an Parametern und somit Konfigurationsmöglichkeiten. Um diese hierarchisch zu organisieren sowie systematisch zu variieren und zu evaluieren, wurde ein entsprechendes Softwaremodul entwickelt und angewendet. Dabei wurde neben dem Root Mean Square Error (RMSE) als zentraler Qualitätsindikator auch der Berechnungsaufwand in einer Weise quantifiziert, die eine Abschätzung der Ausführungsdauer eines Arbeitspakets für verschiedene parallele Rechnerkonfigurationen erlaubt.

Insgesamt wird ein umfassendes Framework für das Monitoring kontinuierlicher Phänomene vorgestellt. Mittels integrierter Erweiterungen zur systematischen Variation und Evaluation wird eine kontinuierliche Effizienzsteigerung der Prozesse ermöglicht.

Umweltmonitoring, Spatio-temporale Interpolation, Sensordatenströme, Kriging-Varianz, Berechnungseffizienz

Contents

Figures

Chapter 1

Introduction

1.1 Motivation and Research Questions

Recent developments in the sector of information and communication technology (ICT) have enormously expanded possibilities and reduced costs at the same time [Gama and Gaber, 2007, Appice et al., 2014]. As a consequence, the monitoring of continuous phenomena like temperature or pollution by stationary sensors has been intensified since its benefits can be utilized at much lower expenses. There are manifold subject areas which are dealing with phenomena that can be modelled as continuous fields [Cova and Goodchild, 2002, Camara et al., 2014]. Analyses based on this specific abstraction model can provide significant benefit to them. The areas of application range from mining, cover matters of geology, oceanology and agriculture and sometimes even touch rather exceptional subjects like medicine or astronomy.

Even more applications can be expected in the future because of the universality of the concept of a *continuous field*. Widely differing types and characteristics of phenomena can be incorporated in appropriate *covariance functions* which express the degree of variability of such a field as a function of spatial, temporal or spatio-temporal distance.

Since a field can never be observed as a whole, it has to be estimated from discrete observations by using some interpolation method. Making the law of variability explicit by the covariance function allows this process to be carried out optimally.

In this context, monitoring can be seen as purposeful organization and processing of observations or samples—the terms are used synonymously here—in order to derive a useful model of a particular phenomenon. The main objective is to provide a sufficient estimation of the phenomenon at arbitrary (unobserved) positions (in space and time) at lowest possible costs.

But in the view of the vast diversity of applications and associated requirements, how should a monitoring be carried out for the concrete case? How can a particular phenomenon be characterized and what consequences does this have for the configuration of the sampling and the whole monitoring process? What are the means to come to well-reasoned decisions in this context? What are the costs for the monitoring? Is there a way to continuously increase quality and efficiency of the monitoring process on a systematic basis? These are the fundamental questions that are addressed by this work.

1.2 Main Challenges and Contributions

Numerous methods can be used to interpolate a continuous field from a set of discrete observations. Generally, there are two principles by which a field can be interpolated from observations: (1) fitting deterministic functions to the observations and (2) assessing their statistic properties and incorporating them into the model. Depending on the phenomenon at hand, a combination of both methods can also be indicated.

This work focuses on the second variant of spatio-temporal statistics or geostatistics (the term is used synonymously here), since its methods are widely accepted and applied and in many cases provide the best results when dealing with continuous phenomena.

The main objectives of this work are (1) to estimate the sufficient sampling density for a given phenomenon and (2) to test different variants of methods and parameter settings of the interpolation. A framework for systematic variation of these factors in order to evaluate them according to several performance indicators is introduced. It is designed for continuous improvement of the overall efficiency of the monitoring process.

In the context of monitoring continuous phenomena there are many challenges concerning the associated tasks of observation, transmission, processing, provision and archiving. There has been and there is continuous progress with respect to increasing hardware performance and decreasing costs. Also, the algorithms associated with monitoring become more powerful and mature.

Many studies exist concerning the processing of concrete datasets of sensor observations in order to derive a continuous field. There is also a vast number of works dealing with the theoretical foundations of geostatisics, although the consideration of spatio-*temporal* modelling is still not very common in this context [Cressie and Wikle, 2011, Gräler et al., 2016].

What is missing in the author's viewpoint is a systematic examination and evaluation of the process of monitoring as a whole. The intention of the framework introduced here is to support an iterative calibration of the used process model. Since diverse performance indicators are provided with each variant of a simulation, the whole process chain can be regarded as “closed loop” where input data and parameters can be related to the output quality [Sun and Sun, 2015, p. 9]. Continuous learning about and improvement of the monitoring process is thus facilitated [Box and Draper, 2007].

The framework presented in this work covers the entire workflow of a simulated monitoring using kriging as interpolation method. Each step of this workflow is listed below. The specific contribution of this work to each particular step is added if present.

1. **Random field generator** The central theme of this work is the investigation of environmental phenomena which can be regarded as continuous in space, time or space-time like temperature, air pollution, radiation etc. The strategy for sampling has to consider the dynamism of the phenomenon and at which level of detail this dynamism has to be captured. So one fundamental question for sufficient observation is how dynamism is related to the minimum sampling density that is necessary to capture it adequately.

   The spatio-temporal dynamism of a continuous random field is controlled by the moving average filter that is used to generate it. All subsequent process steps, starting with sampling, can be tested systematically against changed initial conditions according to this dynamism. The generality of the applied models and approaches can thus be corroborated [Gigch, 1991, p. 62].

2. **Sampling** The critical nexus between the phenomenon itself and its model is established by sampling. The density of the sampling determines at which granularity level the phenomenon is captured. Too sparse sampling can never yield the true character of the observed phenomenon no matter how sophisticated the interpolation is. The geostatistical parameter *range* is an indicator for the dynamism (in space, time or space-time) and therefore also determines the minimum necessary sampling density.

   In this work a formula is deduced from signal processing that estimates the minimum necessary sampling density from given *range* parameter or parameters. The approach is evaluated experimentally. It provides an objective estimation of the necessary sampling density for a given phenomenon and thus makes different observational settings comparable in principle. Another important issue concerning observational data is their efficient transmission and archiving. A compression algorithm is proposed that is designed for this data structure and capable of progressive retrieval.

3. **Experimental variogram** The experimental variogram expresses how sensor observations are actually correlated with respect to their spatial, temporal or spatio-temporal distances. For each possible pair of observations, this distance is related to the corresponding semivariance, which is the squared and halved difference between the measured values. A plot of this relation already conveys an impression of the statistical behaviour of the observed variable with respect to correlation that depends on spatial, temporal or spatio-temporal proximity. The experimental variogram is a prerequisite for subsequent geostatistical analysis.

4. **Aggregation of the experimental variogram** To be applicable for interpolation by kriging, the experimental variogram generated by the previous step needs to be represented as mathematical function. The parameters of this function are fitted to the empirical data. Since the number of points of the experimental variogram grows by $\frac{n^2-n}{2}$ for n observations, the fitting procedure can become expensive even for moderate amounts of data.

   The aggregation of variogram points is one approach to cope with this problem. Such aggregation is usually carried out by a regular partitioning of the region populated by points of the experimental variogram. In this work, the process is carried out with respect to the statistical properties of the point set that is to be partitioned. Different variations of this approach are tested.

5. **Fitting of the theoretical variogram function** With only the aggregated points instead of all variogram points, the fitting procedure can be executed with much less computational effort. The Gauss-Newton algorithm is often used to minimize the residuals of the aggregated points from the function by adjusting its parameters iteratively [Sun and Sun,

2015, Schittkowski, 2002]. By introducing weights, the points representing low distances can be given more influence, which is a reasonable strategy here because bigger distances also tend to be less reliable for parameter estimation due to higher dispersion. Different weighting strategies are tested and evaluated in this work.

In order to make the estimation of optimal parameters by the Gauss-Newton algorithm more robust, starting values for the optimization are deliberately chosen from an n-dimensional grid within quantile borders of each dimension. This alleviates situations where the Gauss-Newton algorithm does not converge or finds several local minima.

6. **Interpolation by kriging** Given the parameters derived from the variogram fitting, the interpolation can be performed at arbitrary positions and therefore also for arbitrary grid resolutions to fill spaces between observations. As a statistical method, kriging provides unbiased estimation of minimum variance [Cressie, 1990, Webster and Oliver, 2007]. Beside the value itself, kriging also provides the estimation variance derived from its position relative to the observations it is interpolated from [Meyers, 1997, p. 464], [Osborne et al., 2008].

   The kriging variance is a unique feature and can be exploited for several purposes. In this work it is used as weighting pattern when merging several raster grid models. The computational effort for kriging can be reduced when subsets of observations are processed and merged sequentially. Merging can also be used to seamlessly integrate new observations into existing models. This is crucial when a (near) real-time model of the phenomenon has to be provided.

7. **Performance assessment** Because the reference model that is observed is synthetic and can be created at arbitrary resolution, the deviation of a model derived from sampling and interpolation can be calculated exactly. When this derived model is provided at same extent and resolution as the reference model, the root mean square error (RMSE) can be calculated easily. This value is the key indicator for the simulation because it expresses the overall quality of the monitoring process [Goosse, 2015]. Algorithmic variants and parameter adjustments will affect this value

and can therefore be used for iterative optimization [Gigch, 1991]. Other performance indicators like computational effort can also be improved this way.

The operational steps listed above constitute the components of a monitoring environment that derives raster grid models from discrete observations of continuous phenomena. It is designed to systematically vary and evaluate different methods and parameter settings of the monitoring in order to iteratively increase efficiency.

Efficiency in this context can be defined as the relation between the expenses necessary to operate a monitoring system and the quality of the model derived from observation and interpolation.

In order to express and systematically evaluate this efficiency, the following aspects of a monitoring scenario need to be quantified:

- extent and dynamism of the phenomenon
- sampling effort
- computational effort
- model quality

These issues above are interdependent. When planning a monitoring system, the first task is to define the extent and to estimate the dynamism of the phenomenon to be observed. The second task is to decide about the necessary granularity and accuracy of the model to be created by the monitoring. Given adequate knowledge about these two conditions, the monitoring system should be designed to sufficiently mediate between them [Beven, 2009, p. 6].

It is up to the decision makers to choose the hardware and software that is appropriate under the given circumstances. The present work is intended to provide methods and tools to support this aim with approaches that can be corroborated experimentally. Continuous efficiency gain concerning the ratio between used resources and achieved accuracy can thus be facilitated. Following the idea of a *closed loop* as also propagated in [Sun and Sun, 2015, p. 9], the guiding principle of this work is depicted in Figure 1.1.

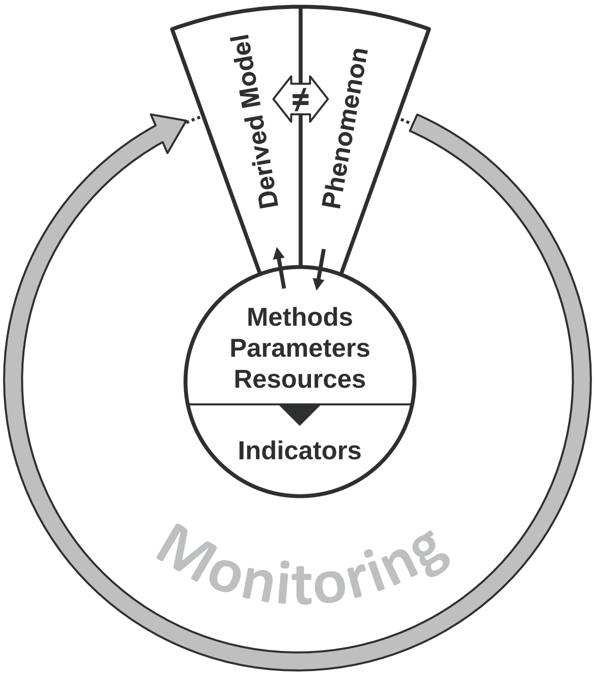


Figure 1.1: General principle of evaluation of monitoring

The schema basically represents the two leading policies that are applied in this work to foster methodological improvement in monitoring continuous phenomena:

1. **Circularity of the monitoring process:** The simulated sampling is carried out on the synthetic reference model. The derived model is generated by kriging interpolation of these samples. When the reference model is given with arbitrary accuracy, as is the case for a synthetic model, the deviation between the derived model and the reference model can be determined exactly.

2. **Systematic variation of methods and parameters and evaluation of output indicators:** The monitoring process as a whole can be configured by various methods and parameters. Namely, these are the density and distribution of observations, the applied interpolation algorithms with their associated settings and also the computational resources used. Variations of these factors will more or less affect the output indicators.

Given this generic framework, continuous improvement of the applied methods and parameters can be fostered. Quantitative evaluation of the monitoring quality and efficiency can be carried out by appropriate indicators for accuracy and computational effort. In combination with a model of the available computing resources, this effort can be concretised in terms of time and energy. Beside storage space, these are crucial constraints especially for large models, (near) real-time systems and wireless sensor networks and should be considered carefully.

The main challenge of this work is to furnish this general framework with methods, parameters and indicators that are appropriate to optimise the task of monitoring continuous phenomena given limited observations and resources.

The remainder of the thesis is structured as follows:

In Chapter 2, the properties of continuous phenomena which are the subject of investigation of this work, are characterized. A general overview of common interpolation methods is also given here.

Beyond observation and interpolation of such phenomena, the process of monitoring entails many technical and also organizational issues that need to be considered in a real scenario. These will be covered in Chapter 3.

The statistical interpolation method used in this work, namely kriging, will be described in more detail in Chapter 4.

On this basis, a system architecture for monitoring continuous phenomena is presented in Chapter 5. It addresses the problems worked out in the previous chapters and is designed to systematically and iteratively improve the efficiency of the monitoring process as a whole.

An experimental evaluation of the proposed concepts is carried out in Chapter 6 before conclusions are drawn in Chapter 7, where also a general perspective on the future development of monitoring systems is sketched out.

Chapter 2

Continuous Phenomena

2.1 Observing and Interpolating Continuous Phenomena

Continuous fields can serve as an appropriate model to describe a variety of phenomena. In fact, most environmental variables are continuous [Webster and Oliver, 2007, p. 57]. Therefore, methods and tools to handle continuous fields are common in many subject areas [Cressie, 1993, p. 11], [Armstrong, 1998, p. 1], [Ma, 2007], [Chiles and Delfiner, 2012]. Some of them are listed below without any claim of exhaustiveness:

- agriculture and soil science
- astronomy
- climatology and meteorology
- ecology (flora and fauna)
- environmental science
- fishery
- forestry
- geology
- hydrology and hydrogeology
- medicine
- mining and petroleum engineering
- pollution control
- public health
- remote sensing
- social geography

The reason for this diversity is that the principle of a continuous field is so universal [Kuhn, 2012, Couclelis, 1992]. Yet, the methods for handling observational data about these phenomena are still evolving. Unlike imagery data where the output product is usually of similar resolution as the observation itself—e.g. when carried out by CCD sensors—, the sensor data that provide discrete values at particular positions in space and time have to be handled differently [Camara et al., 2014, Liang et al., 2016, Couclelis, 1992]. Although the resulting data type—for an easy interpretation by humans as well as by machines—may in fact be discretised as raster grid, this does not at all imply an analogy of the acquired data.

For sensor data registered by regional stations there is a substantial gap to bridge between the original observations and the format required for reasonable interpretation or analysis. This has several impacts on the way such data need to be treated in terms of accuracy, coverage and interoperability:

Accuracy When dealing with grid data from remote sensing, errors can be caused by the sensor itself, by atmospheric effects or by signal noise [Mertins, 1999]. There might also be effects to consider caused by pre-processing the data or resampling it to a different resolution in space and time. Excluding systematic trends, the accuracy of the raster cells is more or less homogeneous. In contrast to that, for a grid derived from interpolated observations of regional stations, the confidence interval will vary significantly depending on the distances of each interpolated grid cell to the observations surrounding it.

Coverage When a region is to be monitored, beside its extent also the observational density has to be considered for both space and time. Unlike for remote sensing [Sabins, 1996], where ground resolution is already an intuitive metric, for interpolation it is necessary to relate the dynamism of the phenomenon to the sampling density (see Section 5.3.2) in combination with the quality and appropriateness of the interpolation method and its parameters (see Section 5.3.4 and 5.3.5).

Interoperability For visualization and analysis that involves other spatio-temporal referenced data, appropriate formats and interfaces have to be provided to access the field data. In this context, a high level of abstraction is a prerequisite for interoperability [Zeigler et al., 2000, p. 30]. Querying a variable at arbitrary positions in space and time can be seen as the most basic function here [Craglia et al., 2012]. It can easily be extended to a grid-based structure which directly supports visualization and analysis. On a more sophisticated level, regional maxima or average values or other types of aggregation can be provided. A general concept for the definition of such queries should precede a syntactical specification of formats or interfaces.

Given these properties of interpolated data, it might appear reasonable to prefer imaging techniques like remote sensing to stationary observations. A coverage of the area of interest by a raster grid that directly reflects the acquiring method and provides homogeneous accuracy is certainly advantageous. Unfortunately, such observations are often unavailable, too expensive or just not applicable to the particular problem. For these situations, interpolation is the only way to provide a gapless representation of the sparsely observed phenomenon. This is the method this work focuses on.

For environmental monitoring, variables like wind speed, precipitation, temperature or atmospheric pressure can be observed by weather stations. The sparsity of the observations according to space and time necessitates reasonable estimations of the value at unobserved positions. There are two general principles for interpolation to provide them: determinism and statistics [Agterberg, 1974, Isaaks and Srivastava, 1990, Webster and Oliver, 2007]. Deterministic approaches align parameters of mathematical functions to observations while statistical ones assume the observed phenomenon as a realization of a random process that is autocorrelated according to spatial, temporal or spatio-temporal distances of observations.

But regardless of the method used for interpolation, the structure of the observational data itself requires particular techniques to make it valuable for interpretation. Whittier et al. suggest the structure of a space-time cuboid on which sliding windows queries can be performed [Whittier et al., 2013]. A spatio-temporal interpolation is performed for each cell that is not covered by an observation. The structure resulting from this process can be interpreted as a three-dimensional grid or *movie*.

A more abstract approach is introduced in [Liang et al., 2016]. A specific data type to manage observational data about continuous phenomena is defined at a conceptional level. The general idea is to store observational data in a standardized way and to select the interpolation method when retrieving the data. So instead of generating and storing interpolated data as additional grid dataset that has to be managed separately, the method integrates observations, interpolation and derived data to one coherent model. Thus, derived grids might be generated immediately with new observations or just on demand when the region is queried. Also mixed strategies are possible here since the management of the data can remain totally transparent to the user or application when using an appropriate query language for field data types.

The vision of such an integrated mechanism for field data is yet far from interoperable realization in available systems, although it appears to be a superior concept. There is still plenty of work to be done in terms of standardisation of naming and implementation of interpolation methods. At least the most frequently used interpolation methods should simply be called out by query parameter and provide identical results from different systems. Since spatio-temporal interpolation is a very complex process with an immense diversity of methods, variants and parameters [Li and Heap, 2008] that is still continuously growing and evolving, this will be a challenging objective. Nevertheless, in the long run it will be necessary to delegate this specific task to a basic infrastructure service component (e.g. a data stream engine [Gama and Gaber, 2007]) to unburden higher-level applications from this complexity. A similar development has taken place for geometries that are stored in database management systems [Brinkhoff, 2013].

Without claiming completeness, a list of commonly used interpolation methods is provided in the next two sections which are named by the most general classification dichotomy: deterministic vs. statistical, or rather, to be more specific, geostatistical methods. Approaches that combine both principles will be covered briefly in the subsequent section.

2.2 Deterministic Approaches

There are various interpolation methods that do not take into account the random character of the observed field and are therefore classified as deterministic [Webster and Oliver, 2007, Li and Heap, 2008]:

Voronoi polygons or Thiessen polygons tessellate a region into polygons so that for each position within a polygon one particular observation is the nearest one. All these positions share the exact value of that observation. As a consequence, there are sudden value steps—discontinuities—at the borders between those polygons, which restricts the scope of application.

Triangulation provides a surface without “jumps” of value by filling the space between three observational points with tilted triangular plates. It has, though, abrupt changes in gradient at the triangle edges.

Natural neighbour interpolation is based on Voronoi polygons. It extends the concept by introducing weights that are proportional to the intersection areas between the Voronoi polygon of the point to be interpolated and the ones of the neighbouring points. In contrast to the preceding approaches, it provides a continuous surface.

Inverse distance weighting presumes that the influence of an observation on the interpolation point is decreasing with increasing distance. This decrease is expressed as the inverse of the distance with an exponent bigger than zero.

Trend surfaces defined by mathematical functions are another way to represent continuous fields. The functions’ parameters are fitted to the observations by regression. With increasing number of observations this approach becomes numerically fragile and the residuals at the observed positions tend to be autocorrelated.

Splines can also be used to create continuous surfaces. They are based on polynomial functions, but there are multiple instances of them which are locally fitted in a way that they join smoothly.

In summary, deterministic in the context of interpolation means that there is some particular law by which the continuum of a value can be *determined*. Just as phenomena that are described by Newton’s physical laws, there is no consideration of randomness [Popper, 2002]. The parameters of these deterministic laws or functions are fitted to actual data, but randomness is not incorporated into the interpolation method. An estimation of variance for the interpolated value can thus not be provided.

Deterministic methods only rarely represent the nature of the environmental phenomenon in a sufficient way. There are usually many complex physical processes involved to produce the particular phenomenon [Webster and Oliver, 2007, p. 47]. Because it is impossible to keep track of all of them, it is often reasonable to regard them as one random process [Isaaks and Srivastava, 1990, p. 196 ff.]. This approach will be discussed in the next section.

2.3 Geostatistical Approaches

In contrast to deterministic methods, geostatistical methods *do* take into account the stochastic nature of the phenomenon at hand. The geostatistical method of *kriging* determines the interpolation value of minimum variance with respect to the covariance structure expressed by the variogram. It should be the first choice wherever the observed phenomenon is, at least approximately, a stationary random process.

Stationarity means that the statistical properties of a process are invariant to translation [Cressie and Wikle, 2011, p. 34]. While *first-order* or *strong* stationarity implies that *all* statistical moments remain constant, *second order* or *weak* stationarity only encloses mean, variance and the covariance function. *Intrinsic* stationarity reduces the conditions to the consistency of the variogram with the data [Webster and Oliver, 2007, p. 268 f.]. Actually, the interpolation of intrinsic phenomena can be carried out using the same kriging system [Armstrong, 1998, p. 90]. Strong stationarity is rather a matter of theory and even weak stationarity is not a prerequisite for kriging in practice [Cressie and Wikle, 2011, p. 323].

Hence, formal geostatistical concepts like stationarity should not be overestimated according to their practical value. Real world conditions only rarely satisfy theoretical considerations and any model "can he considered false if examined in sufficient detail" [Beven, 2009, p. 38].

Nevertheless, with its wide range of variants and parameters kriging provides a sophisticated toolset to adapt to a large variety of phenomena. Within geostatistics, kriging is the most important method, or, as pointed out in [Cressie, 1990, p. 239]:

> The use of the word "kriging" in spatial statistics has come to be synonymous with "optimally predicting" or "optimal prediction" in space, using observations taken at known nearby locations.

Or, as stated in [Appice et al., 2014, p. 51]:

> [...] kriging is based on the statistical properties of the random field and, hence, is expected to be more accurate regarding the general characteristics of the observations and the efficacy of the model.

Its superiority compared over other methods is also emphasized in [de Smet et al., 2007]:

> Of the studies that intercompared methodologies (Bytnerowicz et al. 2002), kriging was objectively shown to give the best results.

The expressive power of kriging has also made it popular in machine learning, where a generalization of the method is known as *Gaussian process regression* [Rasmussen, 2006, p. 30], [Sun and Sun, 2015, p. 351], [Garnett et al., 2010].

In contrast to deterministic approaches, geostatistic methods incorporate the random nature of a phenomenon by introducing the concept of the *regionalized variable*, which is characterized by following equation:

$$Z(x) = m(x) + \epsilon'(x) + \epsilon''(x), \tag{2.1}$$

where $m(x)$ represents the structural component or trend, $\epsilon'(x)$ is the autocorrelated random term and $\epsilon''(x)$ is the uncorrelated random noise [Burrough et al., 2015, p. 172].

Kriging exploits the character of the stationary variable to provide unbiased estimations of minimum variance [Webster and Oliver, 2007, Cressie and Wikle, 2011].

As already mentioned, stationarity, or more precisely, second-order stationarity, implies constant mean, variance and covariance function, or, as Lantuejoul puts it more concretely in [Lantuéjoul, 2002, p. 24]:

- there is a finite mean m independent of x
- the covariance between each pair is finite and only depends on the pair's distance

The covariance function is thus specified as the central geostatistical concept by which the variance between value pairs is expressed as a function of distance. In most cases, this correlation decreases with increasing distance. This explicit rule of autocorrelation is applied when the degree of contribution of each single observation to an interpolation is estimated by an optimal weight. The optimal weight estimation itself is a linear regression problem [Oliver and Webster, 2015] with the associated solution of matrix inversion and therefore of comlexity $\mathcal{O}(n^3)$ [Cornford et al., 2005, Barillec et al., 2011], [Gelman et al., 2014, p. 503].

Whereas the optimal weight estimation is influenced by the observational values themselves, the *kriging variance* is only determined by the covariate structure expressed as covariance matrix at the interpolation point [Guestrin et al., 2005, Garnett et al., 2010]. It expresses the degree of uncertainty or variance that can be expected from the relative positioning of the interpolation point towards the observational points used for interpolation. This *kriging variance* is a crucial information in the context of a setting where (spatio-temporal) autocorrelation is empirically investigated and expressed by the covariance function.

Beside the estimation of uncertainty at a particular position, the kriging variance can be used for sampling configuration and adaptive sampling [Walkowski, 2010, Guestrin et al., 2005, Garnett et al., 2010]. In this work, the kriging variance is used to merge sub-models in order to improve performance or to provide a continuous update mechanism for (near) real-time environments (see Section 5.4.2).

Notwithstanding the sheer overwhelming variety of kriging variants, this work sticks with the basic version of the method known as *simple kriging* [Cressie and Wikle, 2011]. Furthermore, neither noise (*nugget effect*, see [Cressie and Wikle, 2011, p. 123], [Webster and Oliver, 2007, p. 81]) nor variation of semivariance with direction (*anisotropy*, see [Burrough et al., 2015, p. 181], [Cressie and Wikle, 2011, p. 128]) are considered in favour of the more general aim of systematic variation and evaluation of methods and parameters. However, the still rather exceptional aspect of *temporal* dynamism of the phenomenon [Cressie and Wikle, 2011, Gräler et al., 2012, Gräler et al., 2016] is covered by applying the associated spatio-temporal covariance functions (see

Section 4.3).

2.4 Mixed Approaches

On a conceptional level, the dichotomy between deterministic and stochastic methodologies is helpful for a thorough understanding of different approaches. In practice, however, the observed phenomena appear as manifestations of both principles, as pointed out in [Agterberg, 1974, p. 313] for the realm of geology:

> It is important to keep in mind that trend surfaces in geology with residuals that are mutually uncorrelated occur only rarely. More commonly, a variable subject to spatial variability has both random (or stochastic) and deterministic components. Until recently, there were two principle methods of approach to spatial variability. One consisted of fitting deterministic functions (as developed by Krumbein and Whitten), and the other one made use of stationary random functions (Matheron and Krige).

In geostatistics, this issue is today addressed by modelling a deterministic trend, as is the case with universal kriging [Tonkin et al., 2016], [Burrough et al., 2015, p. 186], [Oliver and Webster, 2015, p. 85].

A fusion of deterministic and statistic approaches appears to be a general trend [Chiles and Delfiner, 2012, p. 10]: "The current trend in geostatistics is precisely an attempt to include physical equations and model-specific constraints." Or, as expressed in [Poulton, 2001, p. 192]: "Practical methods may be the joint application of deterministic and statistical approaches."

Generally, the distinction between deterministic and stochastic effects is one of the most fundamental problems of science [Popper, 2002]. In the context of environmental monitoring, however, it is not of decisive importance whether a particular phenomenon is predominantly seen as the result of deterministic or stochastic processes. Rather, the monitoring process should be evaluated by the quality of the model it derives from the available observations.

A simulation environment is a powerful tool to systematically perform such evaluation because it provides full control over the phenomenon model, the sampling and the parameters, and principally unlimited knowledge about the quality and efficiency of the simulated monitoring process, as will be outlined in the next section.

2.5 Simulation

Depending on the subject area, the term *simulation* can have different meanings and therefore different prerequisites. As pointed out in [Pritsker, 1998, p. 31], a simulation is based on a *model*, which is an abstracted and simplified representation of the system under investigation. Predicting the dynamic behaviour of such a model given its initial conditions is then called *simulation*. Likewise, in [Birta and Arbez, 2007, p. 3] the central characteristic of "behaviour over time" is identified or, as expressed in [Banks, 1998, p. 3]: "[the] imitation of the operation of a real-world process or system over time."

Simulation has a wide area of applications. Wherever a system is too complex to be described analytically—which is the case for most systems of interest—, its behaviour can be simulated given the laws and initial conditions. Depending on the goal of the modelling and simulation, a system can be inspected on different levels of knowledge and complexity [Zeigler et al., 2000, p. 13].

The scope of modelling and simulation is by far wider than the one covered by this work. It can be applied to investigate problems of production, healthcare, military, customer behaviour, traffic etc. to name just a few [Law, 2014, Banks, 1998, Zeigler et al., 2000]. The focus here, however, lies on environmental phenomena considered to be continuous in space and time. A thorough reflection of the role of modelling and simulation in this context is given in [Peng et al., 2001, p. 9]:

> A well-tested model can be a good representation of the environment as a whole, its dynamics and its responses to possible external changes. They can be used as virtual laboratories in

> which environmental phenomena can be reproduced, examined and controlled through numerical experiments. Environmental models also provide the framework for integrating the knowledge, evaluating the progress in understanding and creating new scientific concepts. Most importantly, environmental modelling provides the foundation for environmental prediction. Environmental models are useful for testing hypotheses, designing field experiments and developing scenarios.

Within this work, the term *simulation* can be applied to two major issues of the monitoring scenario:

1. The continuous random field representing the spatio-temporal dynamism of the phenomenon
2. The density and distribution of observations carried out on that random field

It could be argued that for purely spatial random fields and the associated observations there is no dynamism at all. Therefore, such a scenario can hardly be called a simulation. On the other hand, the consideration of the temporal dimension would fulfil the prerequisite for a simulation while it would not change the monitoring process *in principle* but just bring in one more dimension. Furthermore, unlike for Monte Carlo methods [Robert and Casella, 1999] which can be assigned to the domain of numerical analysis rather than simulation [Birta and Arbez, 2007, p. 13], continuous random fields are generated to represent real phenomena instead of pure mathematical models. They simulate processes like sedimentation, erosion, diffusion, etc. which in effect are so complex that they can be seen as stationary random.

A more general classification schema for simulation methods is given in [Law, 2014]. It allows for categorization by the following three dimensions:

- static vs. dynamic
- deterministic vs. stochastic
- continuous vs. discrete

This list is complemented with linear vs. nonlinear by [Aral, 2011, p. 44 ff.], which is only relevant for very simple models. Some more elements are added to this list of dichotomous pairs of models in [Jorgensen, 1994, p. 28 ff.]. They are not considered relevant here.

The classification of the simulations carried out in this work is not without ambiguities when using such schemata. So the synthetic continuous random field can be seen as static in time, but only when just spatial dimensions are generated. While the field can be seen as continuous—although discretised to a raster grid—, the observations, being part of the simulation, represent discrete events in space-time. These events can either be carried out deterministically by following an observation plan or stochastically by scattering them randomly in space and time. So it can be said that the concepts and categorizations for general simulation do not necessarily apply to the realm of continuous environmental phenomena.

When shifting to the domain of geostatistics, continuous fields are regarded as realizations of random processes [Cressie, 1993]. Describing such a process by a statistical model and realizing it with a computer is actually called *simulation* [Lantuéjoul, 2002], or, as expressed by [Webster and Oliver, 2007, p. 268]:

> In geostatistics the term 'simulation' is used to mean the creation of values of one or more variables that emulate the general characteristics of those we observe in the real world.

There are generally two variants of simulating continuous fields: unconditional and conditional [Lantuéjoul, 2002, Webster and Oliver, 2007]. When carrying out unconditional simulation, the main interest is to create a random field with properties of a particular covariance function. No further constraints are laid on the realization.

In contrast to that, the idea behind conditional simulation is to create such a random field through a set of real or fictitious observations. Those observations keep their values in the simulated realization whereas for the positions between them, the random values will be generated with respect to the associated covariance function [Lantuéjoul, 2002].

Alternatively, one could also think of just applying kriging interpolation to those observations. But while kriging provides estimates of no bias and minimal variance, the dispersion of the phenomenon is not necessarily represented by it [Webster and Oliver, 2007, p. 271]. So the simulation is to be preferred to interpolation when the overall statistical character of a field is more important than the best possible estimation (no bias, minimal variance) at each position.

Technically, there are several methods to create such random fields. Probably the most popular is the lower-upper (LU) decomposition of the covariance matrix. It has the disadvantage that for n grid cells there is a matrix of dimension $n^2 \times n^2$ to be decomposed. This can exceed computational capacities even for moderate model sizes.

Beside this method, sequential gaussian simulation, simulated annealing and turning bands as are entitled simulation techniques by [Webster and Oliver, 2007].

Dilution, tessellation, spectral and turning bands are listed as methods for generating continuous random fields in [Lantuéjoul, 2002].

The approaches above are either limited in their field of application or do lack cohesion between the generated field and the covariance function.

In contrast to that, the moving average filter provides a flexible and intuitive way to generate a random field representing a particular covariance model. The concept is analogous to spatial filtering in signal processing [Gonzalez and Woods, 2002, Stoica and Moses, 2005]. The filter is applied to a field of independent and identically distributed values (pure Gaussian noise). The output value of each grid cell is a weighted average of the corresponding cell and its surrounding cells in the input grid, whereas the weight is decreasing with increasing distance from the target grid cell. The process is repeated for each grid cell and can be imagined as a moving filter, mask, kernel, template or window [Gonzalez and Woods, 2002, p. 116]. The weighting scheme of such a kernel determines the autocorrelation structure of the output random field and can be derived from an appropriate covariance function (see Section 5.3.1).

Oliver analytically derives kernels for moving average filters for the most common covariance functions (spherical, exponential, gaussian) in [Oliver, 1995]. When applied as filter on Gaussian random fields, those kernel functions produce continuous random fields which are compliant with the covariance functions the kernels were derived for.

In this work, however, there is no mathematical rigorous derivation of the covariance functions used to define the kernel of the filter. Consequently, the continuous random fields that are generated using these functions as kernel filters do not fulfil the conditions of stationarity in the strict sense. Due to the generation process they are, however, random, spatio-temporally autocorrelated and isotropic.

The relationship between the covariance function of the kernel filter and the one of the resulting field is analytically demanding [Oliver, 1995] and beyond the scope of this work. But since real world phenomena do not obey formal statistical considerations either, we neglect this rigour here and focus on the methodological approach for continuous improvement of the monitoring of continuous phenomena as a whole.

2.6 Summary

Continuous phenomena are ubiquitous in our environment and their monitoring and analysis are common tasks for many disciplines. The main challenge is to choose sampling schema and interpolation method in order to generate an appropriate model of the phenomenon. For many natural phenomena there is continuity in both space and time. The dynamism in each of those dimensions should be captured according to the monitoring objectives.

There is a manifold of methods to interpolate between discrete observations of a continuous phenomenon. The geostatistical method of kriging considers the observed phenomena as a random process with a particular autocorrelation structure. Being a powerful method of high adaptivity, it can incorporate complex correlation structures, deterministic trends as well as anomalies like anisotropy. One key feature of kriging is the uncertainty estimation. It is exploited in this work for sequential merging of sub-models.

Regardless of the interpolation method that is applied, it is helpful to consider the idea of a field data type as general abstraction concept [Camara et al., 2014, Liang et al., 2016, Cova and Goodchild, 2002, Couclelis, 1992]. A monitoring system can thus be seen to mediate between *discrete* sensor observations and a *continuous* field that represents the phenomenon of interest.

The discrepancy between this phenomenon and the derived field representation expresses the quality of a monitoring process. With the help of a synthetic continuous random field this performance metric can be used to guide continuous improvement of quality and efficiency of the whole monitoring process.

Chapter 3

Monitoring

3.1 Overview

The monitoring of continuous phenomena like temperature, rainfall, air pollution etc. is carried out by (wireless) sensor networks today. From the perspective of the user of such data, the original discrete sensor observations are not very useful since they often do not cover the spatio-temporal area of interest. So the fundamental task in this context is to provide a gapless and continuous representation of the particular phenomenon either as visualisation in real time or as model for long term archiving, or both. Therefore, it is necessary to cover the area of interest with observations in a way that is sufficient to capture the phenomenon (see Section 5.3.2). The necessary density of observations depends on the dynamism of the phenomenon; for spatio-temporal monitoring this has to be considered for both space and time. From these observations, the value of the particular phenomenon needs to be estimated for unsampled spatio-temporal positions.

Monitoring as a whole can be seen as an optimization problem or trade-off between spent resources and achieved model quality, as stated in [Cressie and Wikle, 2011, p. 26]:

> Looking at this from another angle, the best scientists collect the best data to build the best (conditional-probability) models to make the most precise inferences in the shortest amount of time. In reality, compromises at every stage may be needed, and we could add that the best scientists make the best compromises!

Notwithstanding the fact that sensors and computers become cheaper and more efficient, resources will always remain limited, which constraints the process of monitoring to be as efficient as possible.

Essentially, a monitoring system can be seen as a mediator that uses observations to provide knowledge about the environment that is required by the society. An abstract model of the phenomenon of interest and a monitoring system based on this modelling are the components which mediate between environment and society [Ehlers, 2008].

Figure 3.1 illustrates these very general components of environmental monitoring (environment, model, system, society) with their respective relations and interactions (validation, verification, credibility, monitoring, science, observation, knowledge). Validation, verification and credibility are established concepts in the field of simulation and modelling (see [Law, 2014, p. 246 ff.], also [Banks, 1998, Zeigler et al., 2000]).

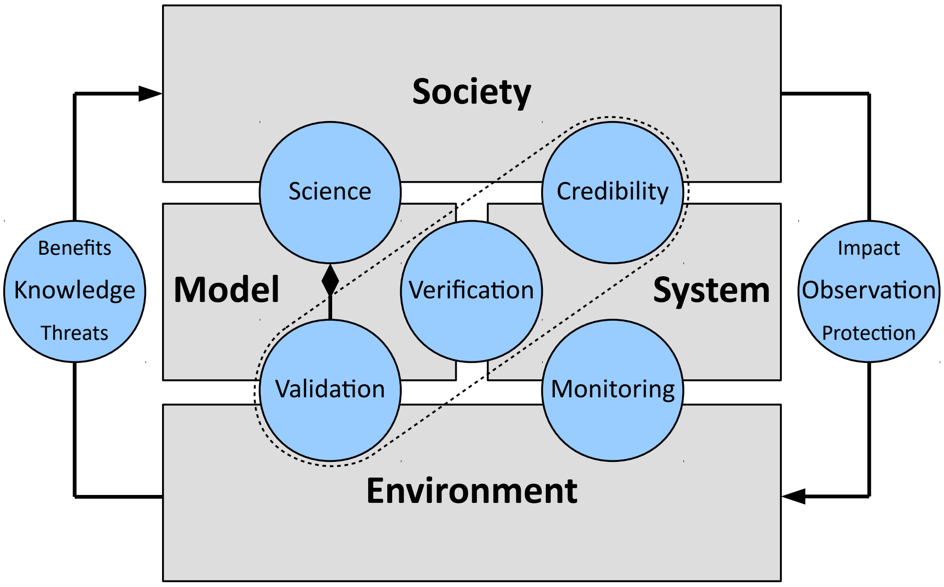


Figure 3.1: Model and monitoring system as mediator between environment and society; validation, verification and credibility (inside dashed frame) as established concepts from simulation and modelling, monitoring as data feeding process, and science as social process developing and validating abstract models. Observation is organized by society in order to gain knowledge, exploit benefits, avert threats, and protect the environment from hazardous impacts

Whereas validation is about whether the right model was chosen, verification explores if this model was implemented correctly. The credibility expresses the degree of acceptance of a particular system solution among its users, consumers or other stakeholders, while science rather discusses and improves the system-independent concepts of the underlying models. Validation as its component contributes to this process.

The superordinate goal of this arrangement is to endow the society with knowledge about the environment to better exploit its resources (benefits) and be protected from potentially hazardous processes (threats). In order to gain such knowledge, the society organizes observations that continuously feed the system with data (monitoring). This knowledge can be used to identify adverse human impact and induce protection policies in a fact-based manner.

This work focuses on continuous environmental phenomena (temperature, humidity, air pollution, etc.) and provides methods and tools that facilitate the iterative development of effective and efficient monitoring scenarios.

On a conceptional or interim level, a continuous phenomenon can be represented abstractly as a field. In order to be easy to interpret by both humans and information systems, the field needs to be discretised to a raster grid of appropriate resolution [Beven, 2009, p. 41], [Cova and Goodchild, 2002], which then becomes the carrier of information about the phenomenon.

Given that the phenomenon, at least in principle, fulfils the criteria of stationarity [Webster and Oliver, 2007, Cressie and Wikle, 2011, Wackernagel, 2003], the geostatistical method of kriging, among spatio-temporal interpolation methods, is often the best choice to derive such a continuous representation from discrete sensor observations (see Chapter 2).

However, to generate such a continuous representation of a phenomenon is a complex calculation process with distinct stages and associated intermediate results. Each of these stages entails several algorithmic variants and parameters that control its behaviour. In a real monitoring environment, the optimal methods and parameters for this process usually remain unknown and can only be estimated by ad hoc heuristics.

In contrast to that, a synthetic continuous random field provides the exact state of the phenomenon at any position in space and time. Therefore, the resulting accuracy of any monitoring process (sampling and interpolation) can always be quantified by the difference between the simulation model and the model derived from the interpolated observations.

The simulation framework described in this thesis was developed to provide an environment to systematically test a wide range of algorithmic variants and parameter settings and inspect their effects on several indicators. Beside the deviation from the reference model also the actual computational effort necessary for each variant can be considered. This makes it possible to quantify and thus compare the efficiency of different approaches. By abstracting the computational effort of a particular calculation from the used hardware it is in principle possible to estimate the expenses in time and energy for any other given computer platform. This can be a critical aspect for wireless sensor networks, large models and environments with real time requirements.

But before inspecting its efficiency indicators, the general requirements of a monitoring system will be listed in the next section.

3.2 Requirements

3.2.1 (Near) Real-Time Monitoring

For many applications it is crucial that the model derived from monitoring is provided in real time or near real time. This is especially the case when the observed phenomenon potentially has severe impacts on security or health like radiation, pollution or heavy rainfall [Aral, 2011]. In all these cases it has to be ensured that observation, data transmission, generation and provisioning of the model is carried out in time according to the requirements.

When the model derived from monitoring is to be provided in real time—e.g. via a web mapping application—one major problem is how to continuously update it. This is especially problematic where observations are irregularly scattered in space and time, as is the case for autonomous mobile sensor platforms. In a sensor data stream environment, one might initially consider two ways to cope with this problem:

1. appending the new observations to the actual set and calculating the model anew
2. creating subsequent subsets of particular size (e.g. 10 minutes time slices) with separate disjunct models per subset

The problem with the first solution is that the number of observations will soon overstrain the computational capacities necessary for model calculation. The problem with the second solution is to choose an appropriate size of the time slice: too short intervals will lead to deficient models due to data sparsity. Too long intervals will on the one hand burden model calculation and on the other hand undermine the timeliness of the provided model.

Another approach is to not completely replace the previous model by the one generated from the newest set of observations, but instead merge this new model with its predecessor. The kriging variance that is—beside the observed value itself—available for each grid cell of the model can be used for weighting when merging two grid models (see Section 5.4.2). A continuous and flexible update mechanism as necessary for a timely monitoring is thus provided.

The merging algorithm also addresses another requirement of real-time monitoring: coping with heavy computational workload when kriging large datasets of observations. When numerous observations have to be processed in near real time, this can become a critical factor even for powerful computing environments. Therefore, appropriate techniques to reduce the workload of the complex task of interpolation [Wei et al., 2015, Pesquer et al., 2011, Umer et al., 2009] are necessary.

3.2.2 Persistent Storage and Archiving

Beside the requirement of (near) real-time monitoring of phenomena, the acquired data, or at least parts of it, will have to be stored permanently for subsequent analyses and considerations of long-term trends. For real-time monitoring it is essential to provide a model that is as accurate and up-to-date as possible under given circumstances and restrictions. The requirements for long-term storage are even more challenging since a good compromise has to be found between the following partly conflicting requirements:

- sufficient quality and density
- small data volume
- originality of the data
- consistency
- informative metadata
- quick and intuitive data retrieval
- interoperability

To be beneficial for as many applications as possible, the data should be stored in spatio-temporal databases with unambiguous spatio-temporal reference systems [Brinkhoff, 2013].

Which data should be stored depends on the type of application that is planned for retrieval. Generally, it is advantageous to keep as much of the original sensor data as possible (originality). So if knowledge that is unavailable at the moment of observation—like a large sensor drift—becomes available after archiving, it can be considered in forthcoming analyses. This is hardly possible if only the *derived data* like raster files are stored.

On the other hand, storing original sensor data means additional effort in the moment of query to provide it in a form that is suited for interpretation or analysis. So if an (n-dimensional) grid of the observed phenomenon is required, the original sensor observations will have to be interpolated according to that grid resolution. Depending on the grid size and resolution and the available computing power, this might significantly delay retrieval.

When efficient retrieval of processed data is of high priority, it might be the best choice to permanently store the data (redundantly) in grid format. There are mature techniques to organize the data management this way. Lossless or lossy compression can help to reduce necessary storage space [Gonzalez and Woods, 2002].

Nevertheless, there is a dilemma according to the management of the monitoring data. Whether to store the original vector data or derived raster data or both has consequences on volume, flexibility, usability, redundancy, consistency, responsiveness, etc. Just as with the process of monitoring as a whole, an appropriate solution has to be a compromise of multiple objectives in accordance with the goals. The concept of a field data type [Liang et al., 2016] is an important milestone on the way to a consistent storage schema for data about continuous phenomena.

3.2.3 Retrieval

For the retrieval of data, there are different scenarios that are reasonable in the context of environmental monitoring. As already mentioned, the monitoring system has the role of the mediator between the available observations and the required knowledge. It fills the gaps in space and time that necessarily remain between the available discrete observations. The overall quality of the model depends on the density of observations and the interpolation method. From this derived model, data retrieval can be thought of in different modes:

1. **Interpolated points:** Values of the variable of interest can be queried at arbitrary positions in space and time. This mode is applied when the variable is needed to examine some critical event.
   For example, when investigating an increased rate of short circuits of a particular model series of power inverters, the actual precipitation at the time and position of each incident might reveal an important hint. Beside the value itself, the estimation variance—as provided by kriging—might also be important to judge the situation.
2. **Interpolated grids:** For visualization or intersection with other data, some equidistant array of values is often needed. So a time series of values for every quarter of an hour could be derived from irregular observations to match the schedule of some other variable to investigate correlations (e.g. air pollution and rainfall). A two-dimensional grid of interpolated values provides a map of the phenomenon at a particular moment in time to be interpreted geographically. Adding the temporal dimension would produce a simulation of the phenomenon as a *movie* [Whittier et al., 2013], as known from weather forecasts.
3. **Aggregations:** In order to get summarized information of a region of particular extent in space, time or space-time, the interpolated grid can be aggregated to the required value. The monthly average value of a pollutant in a particular district is one example of such an aggregation. It presumes an intersection of the interpolated grids (see above) with the (spatio-temporal) target feature. Other indicators like minimum, maximum, median or variance might also represent valuable information. Such aggregations are predestined as an integral element of an alert mechanism within the monitoring system.

As these modes above show, the retrieval of data about continuous phenomena has specific characteristics that are not covered by standard GIS functionality. What is needed for interoperability of systems in this context is a query language that abstracts from the format that the data is actually stored in. It has to provide formal expressions to describe spatio-temporal reference grids, their resolution and extent as well as definitions for aggregations, like monthly average values within a district. With such a query language [Liang et al., 2016], the client does not have to know about the format of the actual data but communicates with the system on a more abstract level.

Additionally, some metadata will also have to be available in order to judge the appropriateness of the data for the given task. Value bounds, mean value and variance might be valuable informations for users of the data. On a more sophisticated level, the variogram can reveal the geostatistical properties of a dataset. Unambiguous identifiers and standardized formats are needed to retrieve and process these metadata.

3.3 Resources and Limitations

The description and prediction of phenomena is the central concern of science [Popper, 2002]. In order to be feasible in practice, only those parts of reality are considered that are relevant for a particular question or task [Birta and Arbez, 2007, p. 6], [Sun and Sun, 2015, p. 9], [Law, 2014, p. 4], [Beven, 2009, p. 17], [Gigch, 1991, p. 91]. Such deliberate reduction of complexity is basically what modelling and simulation is about [Banks, 1998]. According to the intention (knowledge, safety, ecological and economic benefit, ...), the models are designed to answer the crucial questions raised within the particular problem domain.

Monitoring can be seen as as the process that feeds such a model with empirical data in order to align it with reality. The necessary effort for monitoring depends on the requirements according to coverage and accuracy of the model. As for any project, the fact of limited resources will put considerable burden on it. So monitoring strategies will actually be a trade-off between necessary costs and achievable benefits, whereas neither cost nor benefit can always be expressed in monetary units. For the costs, there are following aspects to be considered:

- costs to obtain, install, operate, maintain and depose sensors
- costs for the infrastructure (communication, processing, archiving, provisioning) necessary to keep the monitoring in operation
- human resources (administration, maintenance, adaptation of new technologies, research, cognitive/mental effort...)

On the other hand, there are the various benefits made available through the gained knowledge:

- economic benefits when better predictions result in a more efficient process of added value (like e.g. for fishery, agriculture, forestry...)
- improved knowledge about our environment as basis of existence for present and future generations
- improved quality of life through information (forecasts for weather, pollen drift, air pollution, ...)
- better disaster management (distribution of toxic fumes, radioactivity)
- governmental healthcare

In this context, science and technology can only try to help to explain phenomena, explore causalities, propose solutions, support their implementation and monitor their effects. To provide the necessary resources for this task is the responsibility of society and politics [Beven, 2009, p. 29]. Whether the efforts go along with the proclaimed aims and values should be continuously examined carefully. Science itself has to be rigorous and consistent to withstand being abused by political or economic interests [Walter, 2011, p. 585 ff.], [Jaynes, 2003, p. 19]. Otherwise, it will deprive itself of its legitimation in the long run.

Given these circumstances, the operator or operator team of a monitoring system needs to carefully balance the input resources against the output benefit. In order to facilitate well-reasoned decisions here, two major investigations have to be carried out:

1. Learning about the process and its complexity
2. Identifying the questions intended to answer by the monitoring

The fundamental objective of any monitoring system is to find and establish the link between those two domains in an effective and efficient way. A good compromise between invested resources and derived knowledge has to be found [Beven, 2009, p. 11]. In the following, the means and aspects of monitoring will be listed and discussed according to their limitations.

3.3.1 Sensor Accuracy

The accuracy of each single sensor measurement obviously affects the overall accuracy of the monitoring system. It has to be in accordance with the requirements of the system and should be specified.

Accuracy is a function of bias and precision [Berthouex and Brown, 1994, p. 11 ff.], [Meyers, 1997, p. 60 ff.]. Precision expresses the degree of scatter in the data around a constant value while bias is the deviation of this value from the true value. Precision is associated with random errors while bias is caused by systematic errors.

In the context of monitoring systems there can be mechanisms to detect systematic sensor errors if the data is sufficiently redundant. In such a case, the sensor can be calibrated in order to produce correct measurements. If such a sensor error can be determined to first appear at a particular point in time, all registered measurements since then can be corrected retrospectively. This is hardly possible for derived data like interpolated raster grids, which might be an argument to favour field data types (see Chapter 2 and 7).

Yet, the incorporation of the sensor accuracy into the interpolation process has its own complexity and is not subject of this work. It is assumed that serious sensor errors are detected and considered by other system components. Small errors in the observations are generally assumed to be overridden by the inaccuracy caused by interpolation itself [Parent and Rivot, 2012, p. 9].

3.3.2 Sampling

For the monitoring to be effective, the area of interest has to be covered by an appropriate set of observations. The sampling density and distribution must be sufficient to allow an interpolation at unobserved positions in space and time according to the monitoring objectives. It depends on diverse factors:

- the phenomenon itself and its complexity (e.g. interdependencies with other variables)
- the quality of the monitoring

- the requirements of the monitoring (aggregation, reconstruction, archiving, retrieval, alert, ...)

These factors above are interdependent, which is shown by the following examples:

- The more complex a process is, the more observations are usually necessary to generate a model that adequately represents its dynamism
- The better the physical processes are understood, the less observations will eventually be needed to generate an appropriate model
- Changed requirements according to the aims of the monitoring will probably affect the overall effort that is necessary for the monitoring

The most crucial decision within a monitoring concept is about how the sampling is to be carried out. Insufficient sampling can not be compensated by even the most sophisticated interpolation method. So the region of interest should be covered by enough observations in order to capture the phenomenon sufficiently. On the other hand, within a monitoring scenario it might be the most expensive task to establish, operate and maintain the sensor network. So the other objective is to have as few sensors and observations as possible to achieve the required quality. To find a good compromise here is maybe the most important decision of the operator or operator team.

As already mentioned, there are basically two aspects to be decided for each sampling layout: the number of observations and their distribution. When assuming a regular pattern with constant point distance to observe a particular region, the number of observations can be easily determined. A square grid is often applied, while a hexagonal grid is considered the more efficient variant [Guttorp, 2001, Chun and Griffith, 2013].

A regular pattern, however, can only rarely be applied because sensor sites are subject to geographical and infrastructural constraints (e.g. as for meteorological stations). The sampling pattern of mobile sensors like vehicles or drifting buoys changes continuously. Wherever static or dynamic sampling positions can be assigned freely, a deliberate selection of sampling positions (static) or active sampling (dynamic) should be considered [Osborne et al., 2008, Barillec et al., 2011, Guestrin et al., 2005, Walkowski, 2010].

In this work, however, the sufficient sampling density is estimated for sampling positions that are randomly distributed in the region of interest. It is reproducible, is easy to generate for any dimensionality and produces a big variety of sample distances, which is necessary for robust variogram fitting (see Section 5.3.5). A formula that derives this sampling density from geostatistical indicators will be provided in Section 5.3.2.

3.3.3 Computational Power

Environmental monitoring of continuous phenomena faces limits of computational power according to following tasks:

Data acquisition Mobile units for processing and transmission have limited energy and therefore are also limited in their capabilities

Extensive processing Massive observational data, complex interpolation algorithms and real-time requirements can raise the workload to a critical level

Due to cheaper sensor hardware and new sources like volunteered data acquisition, the availability of environmental observations is continuously increasing [Havlik et al., 2011, Kuhn, 2012]. Consequently, there is an ever growing computational workload for processing and analysis. State-of-the-art computer technology continuously provides more powerful and more energy-efficient machines. In recent years, increasing computational power is less achieved by higher clock speed but rather by increasing parallelization using multiple central processing units (CPU), graphics processing units (GPU) or field-programmable gate arrays (FPGA) [Liu et al., 2012]. This also affects software development since algorithms must apply multithreading techniques to exploit parallel processing architectures [Cormen et al., 2005, p. 772 ff.].

So the challenge for acquisition of environmental data is to use the limited computational resources as efficiently as possible. There are usually several degrees of freedom how to perform a monitoring since the associated tasks of acquisition, processing and transmission can be carried out in different ways [Gama and Gaber, 2007].

For complex calculations that are performed on powerful workstations, servers or even computer clusters, the focus lies on optimizing algorithms, data structures and indexing to achieve sufficient performance for processing and retrieval.

For both scenarios—data acquisition and complex processing—simulation can be used to test and evaluate several variants according to their overall computational efficiency. Therefore, the simulation needs to keep track of the computational work for each scenario (see Section 6.5). This indicator, among others, can be used for iterative optimization according to the prioritized goals.

3.3.4 Time (Processing and Transmission)

For many tasks in environmental monitoring, time is a scarce resource. There are complex analyses that have to be carried out in time in order to be valuable (e.g. pollution alert systems). Sensor observations need to be transmitted and analysed immediately to detect dangerous states and to limit damage [Aral, 2011]. More powerful hardware is one way to address this challenge, but it is not always feasible. Inappropriate costs or limited energy for wireless devices might be arguments against this option.

Nevertheless, the factor time needs to be considered carefully in such situations. So there is good reason to be able to keep track of it explicitly. For a given task, the processing time is basically determined by the workload of CPU cycles necessary for the task, the CPU clock speed and the number of available processors. Compiler optimization, operating system and features of the programming language will also affect performance and should be considered where necessary.

A generic description of temporal effort therefore has to consider two major components: (1) a quantification of the workload of a task by the number of instruction cycles it entails and (2) a formal specification of the performance-relevant properties of the machine the task is processed on. Leaving aside parallelization, the time effort for a particular task is basically determined by the ratio between the cycles of calculation and the CPU clock speed. In most cases however, the proportion of parallelizable code segments and the number of processors of the machine have to be considered as interdependent factors (see Section 5.5).

For transmission, the data rate and the amount of data to be transmitted plus communication overhead will determine the necessary time. Compression and progressive retrieval [Lorkowski and Brinkhoff, 2016] can reduce the transmission time and consequently the energy expense (see Section 5.4.3).

3.3.5 Energy (Processing and Transmission)

Particularly when tasks like observation, transmission or processing are performed on battery-operated devices, the energy consumption of a monitoring scenario has to be considered to achieve an efficient use of resources [Kho et al., 2009]. In a distributed scenario, there are usually several degrees of freedom to fulfil a monitoring task with respect to how and where to perform the several operational steps.

A simple example for that is the exchange of data in a sensor network. One option is to transmit the original data without further processing, the other is to compress the data before transmission and decompress it after receiving it. There is significant computational effort for the compression and decompression process, but since wireless data transmission is much more energy intensive, this method will usually pay off [Appice et al., 2014].

When energy consumption is to be estimated for a particular task, the number of instruction cycles to be processed is the key indicator, just as it was for time consumption. And while the CPU clock speed is taken as denominator when estimating time consumption, we need a factor that quantifies the energy consumption per instruction cycle here (see Section 5.5).

Similar to the estimation of time, the energy expense can be calculated for a given constellation of instructions and hardware specifications. Aspects like parallelization and idle mode energy consumption can make such estimations more complex.

The problem of energy efficiency in wireless sensor networks has been discussed extensively in literature [Gama and Gaber, 2007, Kho et al., 2009, Kolo et al., 2012, Umer et al., 2009, Jin and Nittel, 2008]. When energy consumption is modelled within a simulation framework as described above, it can be determined for different monitoring strategies when processing them in simulations.

This can be of vital importance for wireless constellations.

Data transmission is a critical issue for wireless sensor networks since it usually consumes much more energy than acquisition and processing of the data [Gama and Gaber, 2007, p. 79], [Kho et al., 2009]. As already mentioned, wherever there are multiple feasible scenarios of how to collect, transmit and process the data for a monitoring, the energy efficient variants should be chosen particularly for wireless constellations.

The energy demand for data transmission will depend on data volume, hardware, protocol, medium and geometrical constellation of the network. There is also potential for improvement of efficiency by adaptive configuration of the transmission process [Lin et al., 2016].

Such optimization should beforehand be carried out with help of a simulation, which presumes that the aspect of transmission is adequately modelled according to the issues to be considered like geometric constellation or transmission schedules. A closer examination of this problem is not in the scope of this work.

3.4 Summary

The limitations discussed above are challenging when establishing a system for environmental monitoring. The responsible actors or decision makers need to identify and precisely formulate goals and priorities and to deliberately choose the appropriate devices, methods and configurations to fulfil them. In order to support this complex task, it is helpful to begin with structuring the problem on an abstract level [Gigch, 1991]. The very general properties involved when establishing an environmental monitoring system are: the aims to achieve, the required quality and the generated costs, as illustrated in Figure 3.2.

All these components need to be considered carefully to establish a monitoring system that fulfils the given requirements. Changes in one component usually will affect the other ones, which is indicated by the connecting lines.

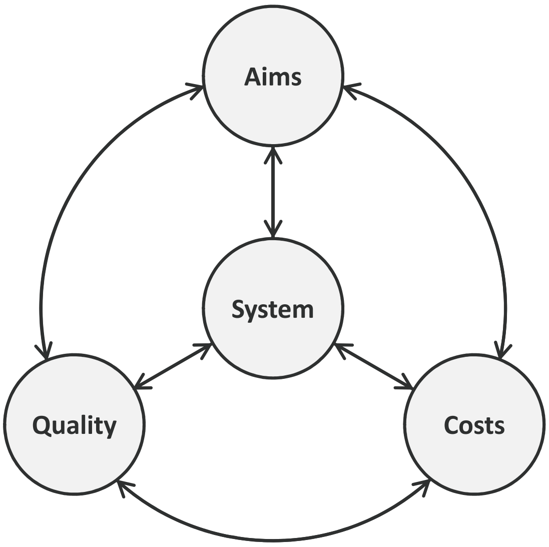


Figure 3.2: Superordinate monitoring system properties and their interdependencies

Each of these superordinate properties entails issues that may or may not play a role for the particular monitoring task. Some of them are listed below for each property:

Aims:

- economic benefit
- scientific progress
- foundations for better planning
- political arguments
- environmental protection
- security and healthcare

Quality:

- coverage
- accuracy
- resolution
- availability
- interpretability
- interoperability (standard conformity)
- response time

Costs:

- infrastructure
- finances
- time
- organizational effort
- cognitive/mental effort
- environmental impact (e.g. stations and their maintenance)

System:

- sensors
- processing units
- (wireless) network
- protocols
- formats
- standards
- methods/algorithms
- parameters
- performance indicators

To bear in mind all of the relevant aspects from the listing above is already an enormous challenge. The interdependencies between those aspects massively increase the complexity in a way that solutions usually can only be found iteratively in an evolutionary learning process [Gigch, 1991, p. 64], [Sun and Sun, 2015].

In this sense, the intention of this work is to support such iteration concerning the aspects of sampling distribution and density, interpolation algorithms and associated parameters.

Depending on the phenomenon and the requirements of the monitoring system, different constellations can be simulated and evaluated by output indicators that express both quality and costs.

The evolutionary process of acquiring new knowledge and improving the model is supported by the circular design or “closed loop” [Sun and Sun, 2015, p. 9] of the framework. By using synthetic data as reference, the *root mean square error* (RMSE) quantifies the fidelity of the derived model and thus is the crucial indicator of the monitoring quality [Goosse, 2015, p. 114 ff.].

Other indicators like computational effort (see Section 5.5.2) are also used. Variations of methods and parameters can easily be processed in batch mode using the concept used in Section 5.5. Given the framework that is described here, it is easy to automatically perform multiple variations of system configurations. The application of this general concept will be outlined in this work after the method of spatio-temporal interpolation is introduced in the next chapter.

Chapter 4

Spatio-temporal Interpolation: Kriging

4.1 Method Overview

The general properties of the geostatistical method of kriging have already been introduced in Section 2.3. It assumes a stationary process (in practice however, only second order and intrinsic stationarity are relevant [Oliver, 1995, Cressie and Wikle, 2011]) and interpolates between observations by estimating optimal weights for them while taking into account their correlation according to their distances. This fundamental relation between the distance and the degree of correlation of two positions is expressed by the covariance function.

When it is applied to an actual set of observations, the method is fundamentally a two-step process [Wackernagel and Schmitt, 2001]:

1. Inspection and mathematical description of the spatial, temporal or spatio-temporal autocorrelation of a given set of observations
2. Interpolation between the observations with respect to the detected autocorrelation structure

One might also say in other words: After specifying the rules of autocorrelation from the given observations, they are applied to estimate the value between those observations. Unlike the deterministic methods listed in Section 2.2, it incorporates the statistical properties of pairs of observations with respect to their spatial, temporal or spatio-temporal relation.

The inspection of the statistical properties is also carried out in two steps: (1) the generation of the experimental variogram and (2) the fitting of the theoretical variogram. Those procedures will be explained in the next two sections.

The autocorrelation structure of a set of observations can be expressed abstractly by the variogram model and the associated covariance function, which has to be fitted to the empirical data. The value prediction for a given point is then performed by estimating the optimal weight for each observation while considering this autocorrelation structure [Oliver and Webster, 2015, Armstrong, 1998].

For simple kriging, the vector of weights λ is determined by:

$$\lambda = C^{-1} \cdot c, \tag{4.1}$$

where C is the quadratic covariance matrix generated by applying the covariance function to each observation pair's distance and c is generated by applying the function to the distances between the interpolation point and the observations, respectively.

The preceding step of fitting of an appropriate covariance function that is needed to populate matrix C and vector c will be outlined in the following.

4.2 The Experimental Variogram

Given a set of observations—often irregularly distributed in space and time—the primary aim of geostatistics is to inspect and describe its statistical properties in order to perform optimal interpolation.

To describe the autocorrelation of a given set of observations, the spatial, temporal or spatio-temporal distances for all possible pairings of observations are related to their *semivariances* by

$$\gamma = \frac{1}{2}(z_1 - z_2)^2. \tag{4.2}$$

Given n observations, the number of pairs p is given by $p = (n^2 - n)/2$. For visual interpretation, for each pair of observations a point can be plotted in a coordinate system that relates spatial (ds) and/or temporal (dt) distances to the respective semivariance (see Figure 4.1).

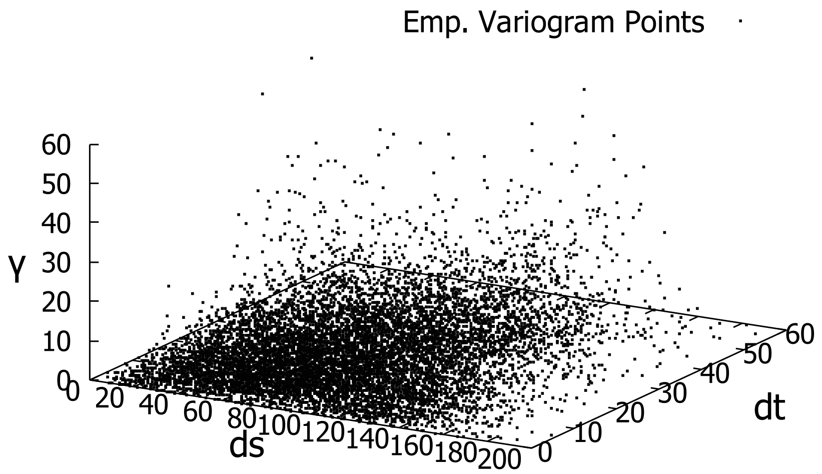


Figure 4.1: Spatio-temporal experimental variogram

The plot clearly reveals the fundamental characteristic of stationary phenomena: observations proximate in space and time tend to be similar in value while distant ones tend to scatter more.

The dimensionality of the variogram (both experimental and theoretical) depends on the number of dimensions that are related to the calculation of the semivariance γ. Taking into account only the spatial distance—be it in one-dimensional (transects, time series) or two-dimensional space—leads to a two-dimensional variogram. The anisotropy—the dependence of correlation not only on the *distance* but also on the *direction*—can be considered by multiple variograms for different circular sectors or, for more precision, a three-dimensional surface [Oliver, 1995, p. 100]. Anisotropy is not considered in this work in order to limit the overall complexity. It could easily be incorporated in both the random field generator and the variogram fitting.

Considering *time* also adds a dimension to the variogram as depicted in Figure 4.1. The autocorrelation structure can be interpreted visually here with respect to spatial and temporal distances. The differing characteristics of correlation decay in space and time and also spatio-temporal interdependencies [Cressie and Wikle, 2011], [Gräler et al., 2012] can thus be inspected. A noticeable scatter near the coordinate origin indicates the *nugget effect* [Webster and Oliver, 2007, Oliver, 1995], which is also not considered here for the sake of complexity.

To be applicable for calculation, the autocorrelation structure that is materialized in the experimental variogram needs to be expressed abstractly as mathematical function. It is called the *theoretical variogram* and will be discussed in the next section.

4.3 The Theoretical Variogram and the Covariance Function

The theoretical variogram can be seen as mediator between the experimental variogram derived from the observational data and the covariance function needed for the population of the covariance matrices. There is a symmetry relationship between theoretical variogram and covariance function, as stated in [Webster and Oliver, 2007, p. 55]:

> Thus, a graph of the variogram is simply a mirror image of the covariance function about a line or plane parallel to the abscissa.

This relationship is apparent when comparing Figures 4.2 and 4.3.

The fundamental geostatistical parameters *sill* (which expresses the dispersion of values for distant points) and *range* (which expresses the distance up to which spatial autocorrelation takes effect) do exist for both representations. Therefore, the fitting of the theoretical variogram to the experimental variogram pointcloud also provides these parameters for the covariance function.

In this context, it is appropriate to point out the fundamental relationship between variance, covariance and correlation, as specified in [Abrahamsen, 1997, p. 9]:

$$c(\tau) = \sigma^2 \rho(\tau), \tag{4.3}$$

where c is the covariance, σ^2 is the variance and ρ is the (normalized) correlation, given the separation vector τ as parameter.

The geostatistical parameter *sill* is sometimes falsely associated with the *dispersion variance*. For a stationary process, the *dispersion variance* is slightly less than the *sill variance* [Webster and Oliver, 2007]. But since it is a parameter to be determined by iterative fitting (see Section 5.3.5), the dispersion variance can very well serve as a priori estimation for the *sill* variance.

Being an abstract model of a (spatio-temporal) dispersion structure (Figure 4.2), it can visually be associated with the experimental variogram (Figure 4.1).

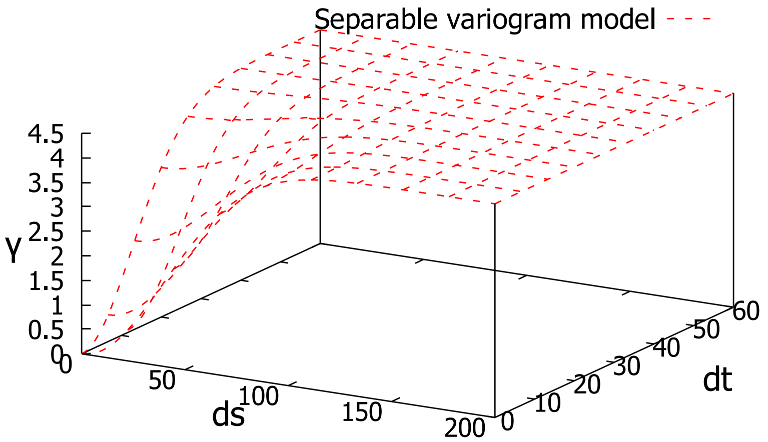


Figure 4.2: Spatio-temporal theoretical variogram

As the mirror function, the covariance function relates the covariance (and thereby also the correlation, see Equation 4.3) to the distance in space and time. As can be seen in Figure 4.3, it has its maximum value at the origin and decays with increasing distance.

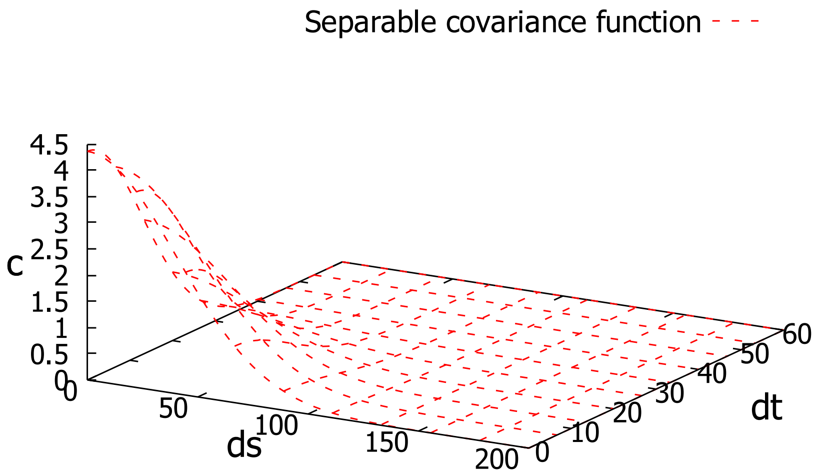


Figure 4.3: Spatio-temporal covariance function

It depends on the characteristics of the phenomenon which variogram model (and therefore which associated covariance function) to choose: How many dimensions (spatial, temporal or spatio-temporal) are there to be considered? By which law does correlation decrease with increasing spatial and temporal distance? Is there some noise for distances near zero (nugget effect)? Does the correlation depend not only on distance but also on direction (anisotropy)? How are the spatial and the temporal dimension entangled [Cressie and Wikle, 2011, Webster and Oliver, 2007]?

The basic two-dimensional representations of three commonly used covariance functions are depicted in Figure 4.4. Their different behaviour especially near the origin and beyond the *range* point implies different characteristics of the corresponding process [Webster and Oliver, 2007, p. 80 ff.]. An initially small (Gaussian) or moderate (spherical) slope that moderately increases (in absolute value) before decreasing again (Gaussian) represents a rather smooth model whereas a steep slope at the origin (exponential) indicates greater dynamic at a small scale. This can be comprehended from the graphs and their associated random fields in Figure 4.4. In contrast to the other two covariance functions, with the spherical the correlation ceases to zero for distances greater than *range*.

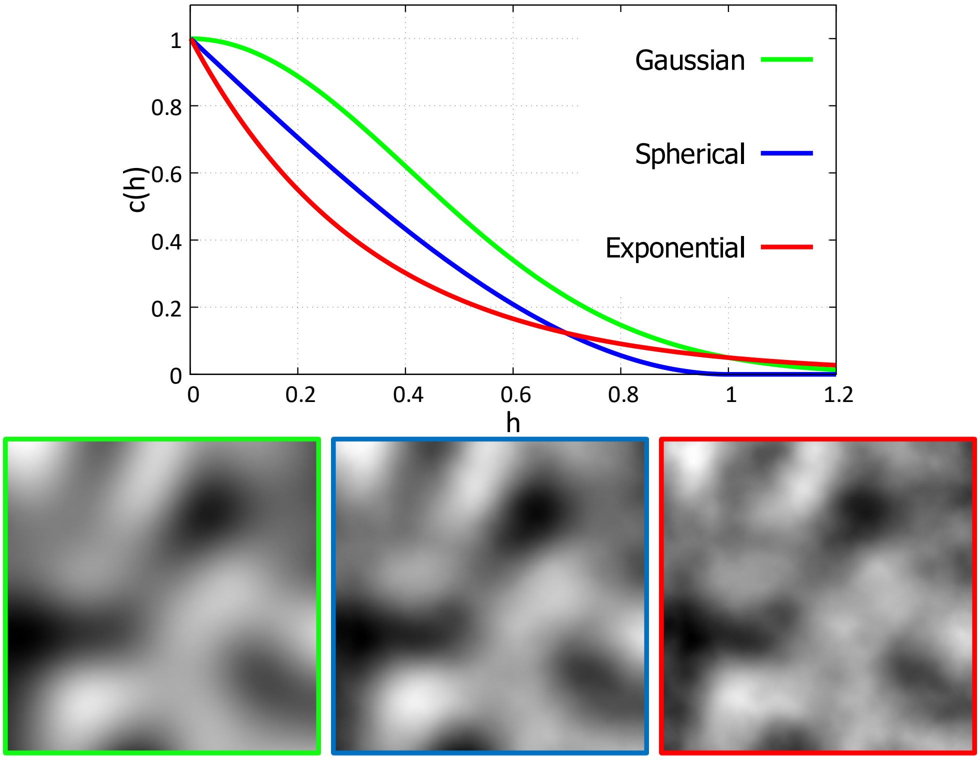


Figure 4.4: Different covariance function types (top) and their respective random fields (bottom) with corresponding frame colors; as can be seen, the smoothness of each field is a function of the slope near the coordinate origin

The respective equations that generate the graphs in Figure 4.4 are given below:

Gaussian:

$$c(h) = s \cdot e^{-\frac{h^2}{\frac{r}{\sqrt{3}}^2}} \tag{4.4}$$

Spherical:

$$c(h) = s \cdot (1 - (\frac{3}{2}\frac{h}{r} - \frac{1}{2}(\frac{h}{r})^3)) \; for \; h < r, \; else \; c(h) = 0 \tag{4.5}$$

Exponential:

$$c(h) = s \cdot e^{-3\frac{h}{r}} \tag{4.6}$$

A graph intersecting the ordinate below the *sill* value (1.0 in Figure 4.4) would represent a noise for near zero distances as *nugget variance* or *nugget effect*. Its representation in the *theoretical variogram*—being the mirror image of the covariance function—is more intuitive since the function starts with value of the nugget variance on the ordinate.

For spatio-temporal kriging, the principle of correlation decay explained above needs to be extended for two input variables, namely spatial and temporal distances. In the context of the simulation framework as described here, this has to be considered for three basic procedures: (1) the variogram-based filter that generates continuous random fields (Section 5.3.1), (2) the fitting of the variogram (Section 5.3.5), and (3) the kriging interpolation (Section 5.3.6).

When more than one dimension is used as explanatory variable of the variogram model, the interaction between the dimensions according to correlation has to be specified (see [Cressie and Wikle, 2011, p. 297 ff.] and [Gräler et al., 2012] for thorough study and derivation of the formulae below). Within the framework, four spatio-temporal variogram models commonly mentioned in literature have been implemented (see Figure 4.5).

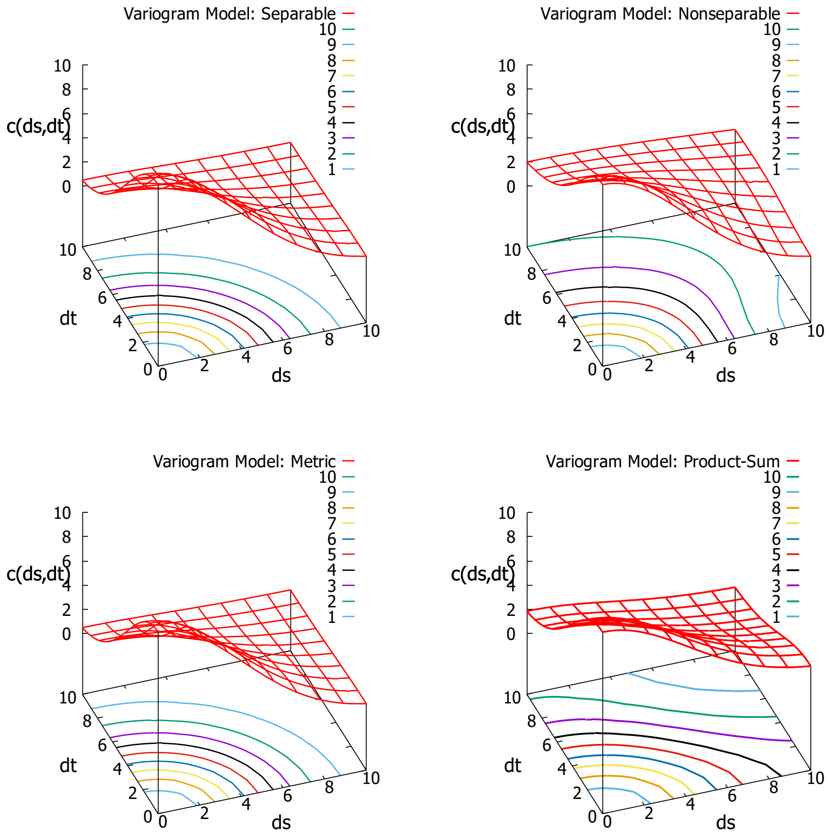


Figure 4.5: Spatio-temporal variogram models: (i) separable, (ii) nonseparable, (iii) metric and (iv) product-sum, plotted as covariance functions with parameters spatial distance (ds) and temporal distance (dt)

In the case of *separable* covariance functions (see Figure 4.5 (i)), the product of two *separate* covariance functions yields covariance values for compound distances in space *and* time with

$$C(s,t) = \sigma^2 f(s)g(t), s \in \mathbb{R}^2, t \in \mathbb{R}, \tag{4.7}$$

where σ^2 is the sill variance, $f(s)$ is the covariance function for the spatial component and $g(t)$ is the one for the temporal component. They might differ in mathematical model and associated parameters to reflect different dynamics in space and time.

In *nonseparable* variants (Figure 4.5 (ii)), the spatial and temporal components are entangled by

$$C(s,t) = \sigma^2 exp\{-k_s^2 \|\mathbf{s}\|^2/(k_t^2 t^2 + 1)\}/(k_t^2 t^2 + 1)^{d/2}, s \in \mathbb{R}^2, t \in \mathbb{R}, \tag{4.8}$$

where k_s and k_t represent scaling parameters for the spatial and temporal component, respectively, and d stands for the number of spatial dimensions [Cressie and Wikle, 2011, p. 317]. The term reflects spatio-temporal interaction that can actually be found in many physical processes [Cressie and Wikle, 2011, p. 309 f.].

The *metric* covariance function (Figure 4.5 (iii)) simply applies a spatio-temporal anisotropy factor (k_t) to align the temporal with the spatial dimension:

$$C(s,t) = \sigma^2 f(\sqrt{\|\mathbf{s}\|^2 + (k_t|t|)^2}), s \in \mathbb{R}^2, t \in \mathbb{R} \tag{4.9}$$

In contrast to that, the product-sum model introduces some more interaction between space and time (Figure 4.5 (iv)):

$$C(s,t) = k_1 f(s)g(t) + k_2 f(s) + k_3 f(t), s \in \mathbb{R}^2, t \in \mathbb{R} \tag{4.10}$$

The equations above express the autocorrelation structure of a random field according to space and time. They are used in this work for generating random fields, choosing and fitting variogram models and interpolating.

In the experimental setup proposed here, the random fields are generated by a filter kernel that applies a separable variogram model. A systematic and thorough investigation of dependencies between the applied variogram *filter*, the applied variogram model and its parameters and the resulting accuracy (RMSE) might reveal interesting dependencies, but is out of the scope of this work. There is, however, a relation between the range value used for the variogram filter and the one determined by the variogram fitting procedure and subsequent kriging that is applied to the random field generated by it: The closer the estimated *range* value is to the one of the variogram filter, the better the interpolation results become (see Section 6.2).

4.4 Variants and Parameters

Kriging has evolved to a complex technique with an almost overwhelming amount of varieties and associated control parameters. Due to this complexity it is often difficult to decide whether it is applied and configured correctly; the mere selection as a method does not sufficiently imply appropriateness or inappropriateness [Meyers, 1997, p. 42 ff.].

An overview of the most used versions is given by the list below without any claim of completeness. It is mainly based on the rewiev in [Li and Heap, 2008]; see also [Burrough et al., 2015, Webster and Oliver, 2007, Cressie and Wikle, 2011] for further study.

Block Kriging In contrast to point-oriented estimations, block kriging (BK) claims for interpolations for (n-dimensional) regions of arbitrary form.

Cokriging Cokriging is the multivariate version of kriging that exploits cross-correlations between different variables (e.g. atmospheric pressure and precipitation) to improve predictions.

Disjunctive Kriging Disjunctive kriging transforms the primary variable to polynomials that are kriged separately and summed afterwards. It is applied when the primary variable does not sufficiently represent a normal distribution.

Dual Kriging Instead of the values themselves, this variant estimates the covariances. It is used when the filtering aspect of kriging is of interest.

Factorial Kriging By applying nested varigrams, the factorial kriging can combine different correlation structures at different scales.

Fixed Rank Kriging This variant is applied for big datasets and reduces the computational workload for inversion of the covariance matrix.

Indicator Kriging When the output variable is supposed to be binary, representing some threshold (e.g. humid vs. arid), indicator kriging can be applied.

Ordinary Kriging Ordinary kriging incorporates the estimation of the mean value by adding lagrange multipliers to the covariance matrix.

Principal Component Kriging Principal component analysis (PCA) is used to identify and quantify correlations in the data, process the identified (uncorrelated) components separately before generating the estimation by linear combination of those components.

Regression Kriging In regression kriging, any trend is determined and removed from the data before the interpolation and added again afterwards.

Simple Kriging Simple variant of kriging that presumes a constant and known mean value. Given the synthetic random fields generated according to this and other statistical parameters, this is the variant that was predominantly used in this work.

Universal Kriging To integrate a trend in the process, universal kriging incorporates a smooth surface as a function of position.

The variants listed above are by no means exhaustive but can only give a hint of the versatility of kriging, which emerges from combinations and subclasses.

Unlike some more intuitive geographic analysis tools (e.g. intersection, buffering, spatial join etc.), kriging as a method requires deeper understanding of the underlying principles to be applied appropriately [Meyers, 1997]. This is also the case for its control parameters, of which the most important ones are listed below.

Sill As already mentioned, the *sill variance* or *sill* expresses the overall variability of a random field. It represents the maximum threshold value for semivariances of pairs of observations (Equation 4.2) and is reached for distances exceeding the *range* parameter (see below). The *sill variance* is not to be confused with the *dispersion variance* which is just the variance of observations in the classical sense. The actual *sill variance* should rather be estimated by fitting the theoretical variogram to the data. However, since this procedure is usually performed by optimization techniques (see Section 5.3.5) the dispersion variance can be a good first approximation.

Range The *range* is the second decisive parameter for kriging since it expresses the distance up to which the observations are stronger correlated than the average of all possible pairs of observations [Webster and Oliver, 2007, p. 89]. In the case of spatio-temporal variogram models, there might be separate *range* parameters for the spatial and temporal dimension. For the separable variogram there is a *range* parameter for the spatial and one for the temporal covariance function (Equation 4.7). In the case of the metric variogram model, the spatial and temporal distances are cumulated by using an anisotropy factor (Equation 4.9). For the nonseparable variogram (Equation 4.8) and the product-sum variogram (Equation 4.10) there are factors to control scaling *and* interaction of space and time [Cressie and Wikle, 2011].

Nugget Effect Just as for the range, also the nugget effect might show different dynamics in space and time, resulting in a joint short-distance noise. The nugget effect, however, is often difficult to estimate in practice since the measuring stations are chosen at distance to avoid redundancy for economic reasons [Gräler et al., 2012]. As mentioned before, the nugget effect is not considered further in this work.

The parameters above are usually estimated individually for each dataset by fitting the theoretical variogram to the respective experimental variogram. The more parameters take part as variables in this fitting procedure, the more cumbersome the optimization can become. This aspect will be addressed in Section 5.3.5 and Section 5.5.

Given the variety of methods and parameters mentioned above, it is worth considering an architecture that provides the interpolation of a value of interest as a service. Without having to deal with too many details and program specifics, a common method with approved configuration could simply be identified by a unique name. Alternatively, for more flexibility the service could be configured by an appropriate interface (see Chapter 7).

4.5 Kriging Variance

Apart from the advantages of the method of kriging that have been covered so far, the provision of the estimation variance is unique among interpolation techniques [Oliver and Webster, 2015, p. 1, p. 60]. It is a by-product of each point interpolation and reflects the uncertainty of estimation resulting from the constellation according to spatial and temporal distances to observations. Just as the estimated variable itself, it represents a continuous field that can be discretised as raster grid.

The kriging variance is derived by multiplication of the weight vector from Equation 4.1 with the vector c that contains the covariance values of each observation with respect to the interpolation point:

$$v = \lambda^{\intercal} c \tag{4.11}$$

As the ingredients of the derived value show, it only depends on the covariance structure that is given by the geometric constellation of the observations and the interpolation point; it does not depend on the observed values themselves [Guestrin et al., 2005].

When creating raster grids by interpolation with kriging, it is useful for many purposes to also store the kriging variance (or deviation) for each cell as additional dimension or channel. This "map of the second kind" [Meyers, 1997] reflects the confidence of estimations as a continuum with respect to the proximity to observations.

In the scope of this work, the kriging variance represents a highly valuable information. In the context of monitoring it—or its complementary value—can also be interpreted as *information density* and thus be exploited to address several problems of monitoring:

Continuous integration of new observations In monitoring scenarios where a state model has to be provided in (near) real time, there is the problem of how to seamlessly integrate new incoming observations. For workload and consistency reasons, this updating should be carried out without having to (1) calculate the model anew using all previous *and* the new observations or (2) replace the old model by one relying only the most recent but probably too few observations. A compromise would be a sliding window [Whittier et al., 2013] containing only observations that do not expire a particular actuality. But depending on the spatio-temporal distribution of the observations and the size of the time window, the approach might cause temporal discontinuities if the window is to small and heavy computational workload if it is too large.

Alternatively, the model can be updated smoothly and selectively wherever new observations occur. To accomplish this, the kriging variance—continuously available for the predecessor and the new model—can be used as weighting schema by which both models are merged (see Section 5.4.2). This method is highly flexible in terms of the number of new observations to be integrated because it retains the previous model where no new information is given instead of indifferently overwriting it.

Performance improvement by model subdivision Apart from its application for continuous updating in a data stream environment as described above, the method can also be used to mitigate the computational burden of numerous observations. Instead of including all observations in one large model, it can be divided into subsets that are processed individually and then merged by weights based on their kriging variances (see Section 5.4.2, also [Lorkowski and Brinkhoff, 2015b]).

Confidence about critical state checks In a monitoring scenario with critical state checking (see Figure 5.13, p. 100), the kriging variance can significantly help to put such a statement on a objective basis. So if a sensor network is installed to push an alert in case of some exceeded threshold, the *intrinsic* idea behind it is to permanently exclude the possibility of that threat. Whether this is the case because of an actually exceeded threshold or because some sensors are down and therefore no sufficiently secure knowledge is available: some actor needs to be notified to induce some predefined procedure. The failure of sensors might eventually not change the derived *value* itself, but rather its *variance* and therefore the confidence of the associated state check.

Adaptive filtering Data sparsity is a very common problem for monitoring scenarios. However, with an increasing number of available low-cost sensors just the opposite can become a problem that calls for decimation of observations. It should be carried out deliberately since autonomous mobile sensors might not be distributed homogeneously (like drifting buoys, see [Wei et al., 2015]). Observations that would only minimally contribute to updating the model should preferably be left out. The kriging variance map as indicator for *information determination* or *information density* can be used for such adaptive filtering: only observations in regions above a particular variance threshold are considered in order to limit data redundancy. Another way would be to order a set of new observations by the values determined by their position on the kriging variance map to leave out a particular number or quantile of the data. The utilisation of the kriging variance as filter provides a flexible and adaptive solution wherever too much observational data is a problem.

The various areas of beneficial application of the kriging variance or *kriging error maps* [Meyers, 1997] as listed above constitute a strong argument in favour of kriging as interpolation method. The estimated confidence for each interpolated value is such a crucial information that it should always be considered carefully.

4.6 Summary

Notwithstanding the computational burden kriging lays on the monitoring system, it offers several features that make it unique compared to other interpolation methods:

- It is an unbiased estimator of minimum variance [Oliver and Webster, 2015]
- It is well established in geosciences, but also in the area of machine learning, where it is known as *gaussian process regression* [Rasmussen, 2006, Gelman et al., 2014]
- By the concept of the variogram and the associated covariance function, kriging allows to consider even complex correlation structures with respect to time, space, space-time, periodicity, nugget variance, anisotropy etc. [Webster and Oliver, 2007, Cressie and Wikle, 2011]. Given this powerful feature, the method is capable of adapting to a large variety of phenomena
- There is a vast number of kriging variants to address the wide range of problems associated with the monitoring of continuous phenomena [Li and Heap, 2008, Burrough et al., 2015], [Meyers, 1997, p. 43]
- The parameters of the variogram are usually estimated from empirical data; they specify the statistical properties of a particular phenomenon; their values might provide valuable information for retrieval when provided as metadata
- The kriging variance with the associated kriging error map or the “map of the second kind” [Meyers, 1997, p. 464] is crucial where confidence of the interpolated values is important. It can also be exploited for features like performance improvement, continuous seamless updating and filtering (see Figure 5.13, p. 100)

The undisputable high computational burden ($\mathcal{O}(n^3)$ for the inversion of the covariance matrix) of kriging may disqualify the method where high throughput goes along with real-time requirements. In such cases, inverse distance weighting (IDW) might be preferable due to its lower complexity [Whittier et al., 2013].

On the other hand, its often superior interpolation quality [Appice et al., 2014, p. 51], the explicitly estimated and intuitively interpretable statistical parameters, and the very valuable additional information of the kriging variance make it a choice that should always carefully be considered. For operating an environmental monitoring system it provides sophisticated means to address many problems that occur in this context. With its diverse variations and parameters it is well suited for iterative improvement within a simulation environment.

Chapter 5

System Architecture

5.1 Overview

On a very abstract level, the problem addressed within this work can be expressed as illustrated in Figure 5.1: a continuous phenomenon with its specific dynamism in space and time is observed by a set of measurements of particular density and distribution. From these discrete observations of the phenomenon, a continuous model can be derived by applying an interpolation method. This model needs to be discretised for interpretation or analysis. A regular grid of appropriate (spatio-temporal) resolution is much easier to interpret and analyse than the original dispersed observations.

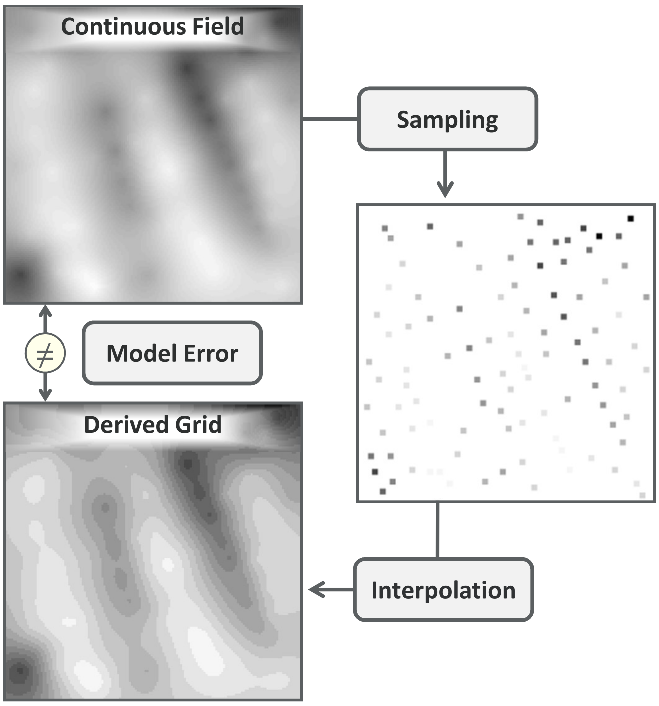


Figure 5.1: Monitoring principle for continuous phenomena

In the context of a *simulation framework* with a synthetic continuous field as phenomenon model—usually realized as a grid—there is the advantage to be able to compare this reference with the model derived from the monitoring. The two main processes of monitoring, namely sampling and interpolation, can thus be evaluated by a meaningful quality indicators like the root mean square error (RMSE) as difference between the synthetic model and the interpolated model. By varying methods and parameters of the processes and observing the effects on this quality indicator, the monitoring can iteratively be improved [Barnsley, 2007, p. 18 f.], [Goosse, 2015, p. 114 f.].

Beside the quality, also the efficiency of the monitoring can be judged by introducing indicators for the effort for computation and eventually also for data transmission. By improving methods and algorithms, the expenses in time and energy can eventually be reduced while achieving similar quality of monitoring.

As illustrated in Figure 5.1, the main goal of the framework is to allow for continuous improvement of the entire process of monitoring according to accuracy and efficiency.

The framework presented in this work addresses this goal by (1) creation of continuous random fields and simulation of monitoring, (2) systematic variation of the interpolation method and their parameters (3) evaluation of the process variants using different performance indicators.

These concepts and tools will be presented in the rest of this chapter. Their experimental application and evaluation is carried out in Chapter 6.

5.2 Workflow Abstraction Concept

The area of concern of this work is the acquisition and interpolation of environmental, spatio-temporally referenced observational data (monitoring), the processing of such data (analysis) and the modelling and execution of different variants of these two activities (simulation).

These tasks necessarily include the management and processing of spatio-temporally referenced data, which is often computationally intensive. It is therefore crucial to find working solutions under limited resources, especially for battery operated systems like wireless sensor networks. Beside the efficiency aspect concerning computation time, energy and data volume, most of all the *quality* achieved by the applied interpolation method is a crucial evaluation metric. It can be used to evaluate several methods and adjust the corresponding parameters to generate best solutions.

In simulated scenarios where the synthetic model provides full knowledge about all relevant environmental parameters, it is easy to determine the quality of the monitoring of a continuous phenomenon by comparing the reference model with the one derived from interpolated observations. In this case, the root-mean-square error (RMSE) is the target indicator to be optimised.

In distributed environments, the transmission of data is often a critical aspect because it is relatively energy intensive. So compression and decompression of data in this context is an important issue (see Section 5.4.3). This is also the case for long-term archiving in databases, where also an appropriate indexing strategy is indispensable especially for spatial, temporal and spatio-temporal data for efficient retrieval [Brinkhoff, 2013, Rigaux et al., 2001, Samet, 2006].

On an abstract level, the considerations above can be condensed to a data process model as sketched in Figure 5.2. The model entails the process itself, the input and output datasets, and properties associated with all of those elements.

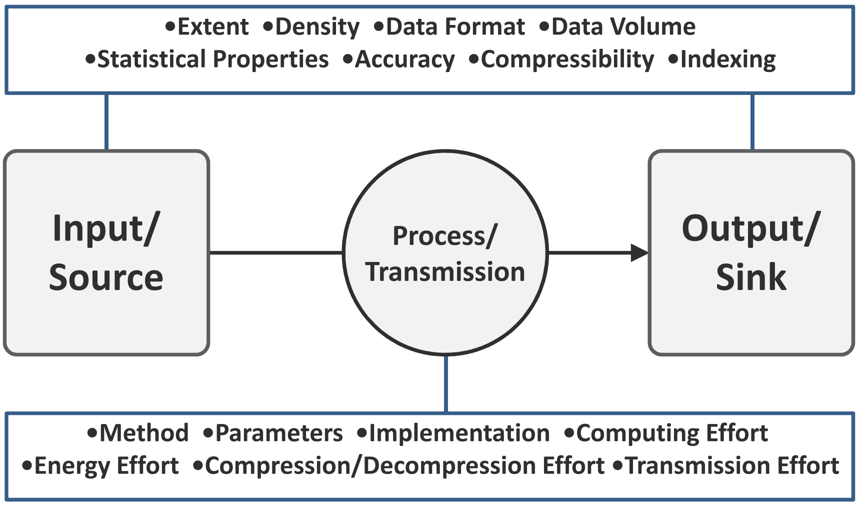

•Extent •Density •Data Format •Data Volume
•Statistical Properties •Accuracy •Compressibility •Indexing
Input/ Source
Process/ Transmission
Output/ Sink
•Method •Parameters •Implementation •Computing Effort
•Energy Effort •Compression/Decompression Effort •Transmission Effort

Figure 5.2: Abstraction of a process/transmission step with associated properties

The term *component* is used by [Taylor et al., 2009] for a processing unit and defined as follows:

> A software component is an architectural entity that (1) encapsu-

> lates a subset of the system's functionality and/or data, (2) restricts access to that subset via an explicitly defined interface, and (3) has explicitly defined dependencies on its required execution context.

In the context of the system introduced here, a component's dependencies to the entire system are given by the input data, the output data, the parameters that control its behaviour and the resources necessary to execute it. A complex simulation will be composed of multiple such process steps or components sequenced by their logical order (see Figure 5.3).

Basically, Figure 5.2 entails the generic properties of input and output datasets (source and sink for transmission processes) that affect such a process step. The process itself is determined by its concrete realization (method, implementation, parameters) and the input dataset. To evaluate the quality and efficiency of the process or the transmission, respectively, the indicators for expense in computation, energy, compression/decompression and transmission are identified.

As will be shown in Section 5.5.2, the computational cost for a particular workload can be expressed in terms of time and energy by assigning a specific hardware configuration.

From the data perspective, we find the properties extent in space, time and value, the amount and distribution (point data), the resolution (raster data), and their format and its compressibility. Statistical properties can help to decide whether the data can be used for the intended purpose. In simulated scenarios as in this case, it is also possible to exactly quantify the accuracy of the entire monitoring process by indicators like the RMSE. In order to enhance the performance of data retrieval, an indexing can be attached to the data.

With regard to complex computing systems for monitoring, analysis or simulation that need to work with limited resources (computation capacity, time, energy), such abstraction is necessary to evaluate scenarios with respect to different hardware configurations, algorithmic methodologies, corresponding parameters and balancing of workloads in distributed environments.

Having (near) real-time and/or mobile monitoring applications in mind, the factors computation workload, data volume and compressibility (see Section 5.4.3) gain more importance. Given the aspects associated with each process step as depicted in Figure 5.2, a careful balancing of these partly interdependent factors is essential to address both the requirements and the restrictions of a monitoring environment.

The properties of the abstraction model as sketched above will be discussed more thoroughly in the following two subsections.

5.2.1 Datasets (Input/Source and Output/Sink)

There are several generic properties of datasets that appear relevant in the context of a monitoring environment, as shown in Figure 5.2.

The extent of a dataset defines its spatial and temporal expansion in a global reference system. It is the crucial criterion to organize extensive environmental data. Without specific indexing techniques, it would not be possible to provide efficient retrieval of the data [Rigaux et al., 2001, Appice et al., 2014].

The frequency and distribution of observations define the data density for vector data, the resolution expresses this property for raster data, respectively.

The syntactic structure of each dataset is determined by its data format. The underlying model reflects the level of abstraction [Gigch, 1991, p. 69] of the described phenomenon.

The data volume that is necessary for each dataset depends on the data format and the number of features (vector) or on the extent, resolution and color depth (raster), respectively.

Compressibility is the ratio by which the data volume can be reduced by applying a compression algorithm. It can produce lossless or lossy representations for both raster [Gonzalez and Woods, 2002] and vector [Huang et al., 2008] data formats.

Statistical properties are of high value when reviewing and analyzing datasets [Gama and Gaber, 2007]. Classical aggregates like mean and variance provide basic characteristics of the data. More sophisticated indicators like a geostatistical variogram convey deeper knowledge about the general structure of the data. This knowledge can be exploited by applications or users to decide whether a particular dataset has to be considered at all.

The accuracy of a dataset that represents a field and was generated from observations can be derived by different methods. A root-mean-squared error (RMSE) can usually only be calculated when some reference is given, as is the case for simulations. Cross-validation is often the method of choice for empirical data where the only available knowledge consists of observations themselves, although it does not necessarily have to be a good accuracy indicator in every case [Oliver and Webster, 2015, p. 68].

Spatio-temporal indexing of a dataset is the prerequisite for efficient data retrieval [Brinkhoff, 2013]. For observational data (vector), the conventional method of defining indexes per feature table might be used. However, the management of these data on the granularity level of single observations might add too much overhead, especially when considering long term archiving.

Instead, it appears reasonable to conflate spatio-temporal areas of observations and exploit their proximity of coordinates and observed values for compression (see Section 5.4.3). The spatio-temporal indexing would refer to those conflated sets which then have to be decompressed on demand. When configured appropriately, this overhead should be outweighed by the benefit of less storage space.

5.2.2 Process/Transmission

On an abstract level, a process step generates an output dataset from an input dataset by applying an algorithm with associated methods and parameters (see Figure 5.2).

Limited resources like computation power, time and energy put considerable demand on the processes to be as efficient as possible. There are generally two different modes of improvement: (1) optimising procedures that are sharply defined according to their result (e.g. sorting of a list) and (2) optimising procedures that are only vaguely defined (e.g. interpolation of observations, fitting of a variogram). In the second mode, there is always a trade-off between cost and effect of a particular procedure that might be difficult to weight. It is this mode that the present work focuses on.

A thorough analysis of requirements, realistic workloads, appropriate hardware and feasible variants of transmission and processing is necessary to evolve the monitoring towards more and more efficient solutions. Especially environments with wireless communication, big datasets and/or real-time requirements put considerable constraints on the way a process is executed.

A continuous overall optimization requires both the evaluation of the quality of the resulting interpolated model, often indicated by the RMSE, and the tracking of workloads according to transmission and computation (see Section 5.5.2). These have to be registered for each process step and summarized in order to weigh the quality of a monitoring against its costs (see Figure 3.2, p. 43).

For an extensive experimental study which compares various configurations, it is helpful to carry out these variations in a systematic and automated way. Especially when there are manifold methodological and parametric settings that need to be tested and evaluated (see Table 6.3, p. 168), such an approach can become indispensable for reasonable investigation.

In the context of a complex monitoring scenario as introduced here, there are generally two modes of variation, which can be related to different scales of measure [Cova and Goodchild, 2002], [McKillup and Dyar, 2010, p. 16]:

1. Switching between algorithms and different implementations (nominal scale)
2. Adjusting a parameter (ordinal, interval, and rational scale)

For the first mode, switching between different variogram models (exponential, spherical, Gaussian) is an example of its application to a simulation scenario (see Section 6.2). The second mode can be used to vary the number of observations by defining a minimum, a maximum and an increment value (see Section 6.1). This mechanism can also be applied to floating point parameters that are not included in the Gauss-Newton optimization (see Section 5.3.5).

The circular design or "closed loop" [Sun and Sun, 2015]—as described in Section 1.2 and depicted in Figure 1.1—facilitates continuous optimization of monitoring by processing multiple simulation scenarios with different conditions according to sampling design, algorithms and parameters.

Such an optimization can be carried out with respect to several target indicators, of which the following are of central interest with respect to algorithmic improvements:

- quality (e.g. quantified by RMSE)
- logical computational workload (instructions)

- physical computational workload (time and energy)

Whereas the quality indicator RMSE is rather straightforward in the scenario that is regarded here, the computational workload can be either regarded from a *logical* or a *physical* perspective. This differentiation is necessary when the execution effort is to be estimated for different hardware environments. The general concept is introduced in Section 5.5 and is applied experimentally in Section 6.5.

5.3 Monitoring Process Chain

In the last section, the properties of process steps, input and output datasets and performance indicators have been set out. In combination with a solution for parameter variation, a generic toolset for systematic improvement of monitoring is provided.

In this section, each step within a monitoring scenario is specified according to its methodology, parameters, input and output data. An overview of this process chain is given by Figure 5.3.

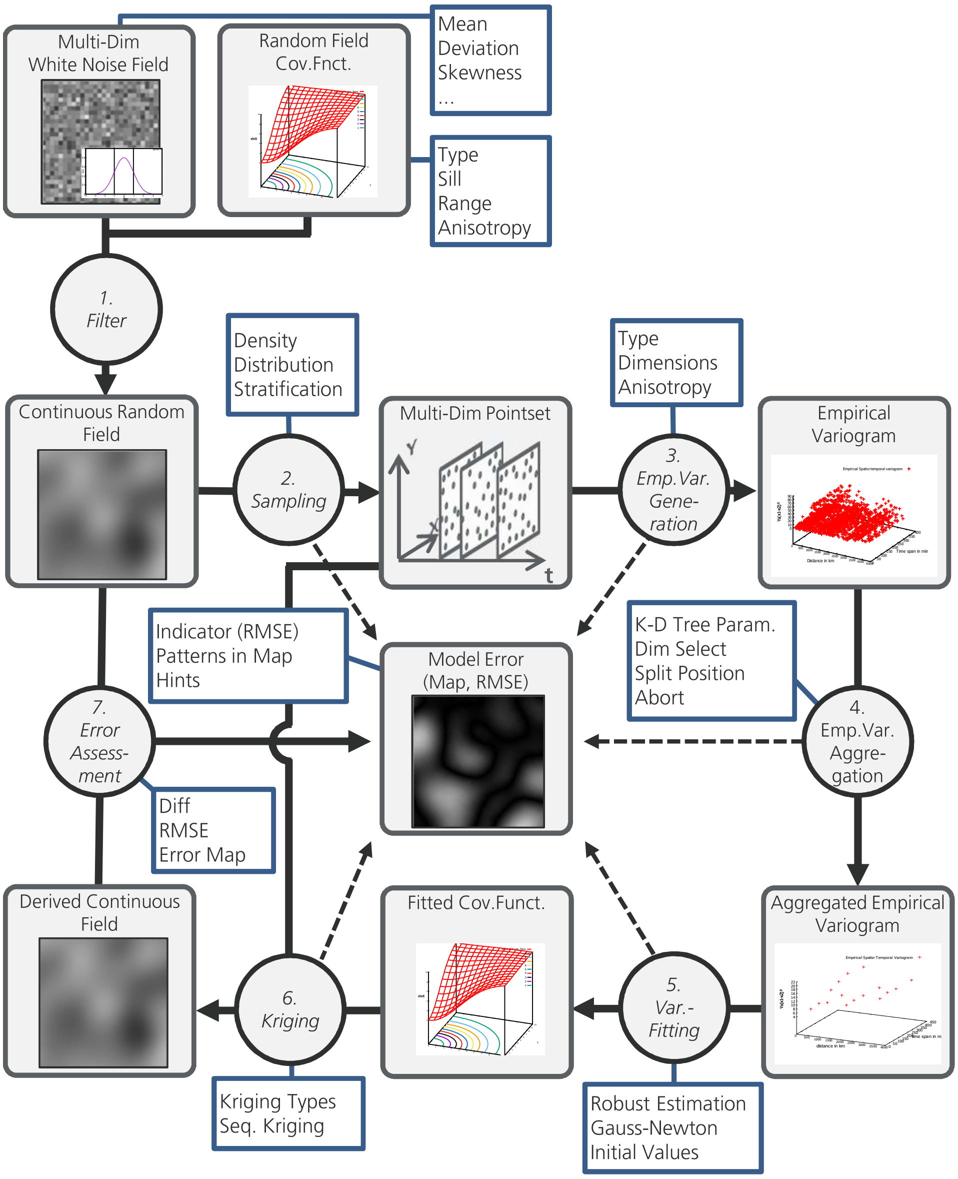


Figure 5.3: Simulation framework architecture with datasets/models (rounded boxes), processes (circles) and their parameters (blue boxes), and their impact on the model error (dashed arrows)

The main objective in the simulation scenario illustrated by Figure 5.3 is to identify those methods and parameter settings that yield the smallest RMSE (7) and therefore the best approximation of the continuous random field generated by the filter (1). For the sampling of this random field (2), for the generation of the experimental variogram (3), for the aggregation of the latter (4), for the variogram fitting (5), and finally for the interpolation by Kriging (6), there are multiple variants of algorithms and associated parameters to be evaluated.

The proposed simulation framework was implemented using programming language C# [Nagel et al., 2005]. With the GNU project *gstat* [Gräler et al., 2016], there already exists a powerful package for geostatistical processing. It is implemented in the statistics-centric programming language *R*.

For the simulation framework that is referred to in this work, the full-featured programming language C# was preferred due to its expressiveness through the support of multiple paradigms, its mature state and wide support, and its portability to almost all platforms. Although this decision means "reinventing the wheel" in many respects, it provides maximum independence and flexibility according to modelling, optimization, portability and interoperability.

In the following subsections, each step of the process chain of Figure 5.3 is specified in detail.

5.3.1 Random Field Generation by Variogram Filter

In order to evaluate different variants of monitoring continuous phenomena, a continuous field is generated as reference model on which sampling and interpolation is carried out. Because continuity is only a theoretical concept, the field has to be discretised in some form. A regular grid raster as most common representation of such data structures is also used here.

Beside the two-dimensional grid raster that can easily be visualized as greyscale image, also three-dimensional fields are used to represent models that include the temporal dimension. Such a model can then be visualized as a sequence of images or a *movie* [Whittier et al., 2013].

These fields are considered, at least approximately, stationary, which means that their statistical properties mean, variance and autocorrelation are invariant under translation in space and time [Cressie and Wikle, 2011]. In the strict sense, however, stationarity is a concept that can only occur in fields of infinite extension (see Section 2.3, also [Webster and Oliver, 2007]). But since natural phenomena cannot fulfil this criterion either, the data generated by the filter is considered to be sufficiently stationary for the purpose of simulated monitoring.

Pure White Noise Grid As a prerequisite for a multidimensional continuous random grid that has to be generated, a grid of pure white noise of the required dimensionality and resolution is created. Its grid cells are independent and identically distributed (IID) random values. For the fields generated here, it is characterized by normal distribution, the preset mean value μ and the standard deviation σ. In order to create normally distributed variables from uniformly distributed pseudo-random numbers, the well-known Box-Muller algorithm [Robert and Casella, 1999, Press et al., 2007] is used.

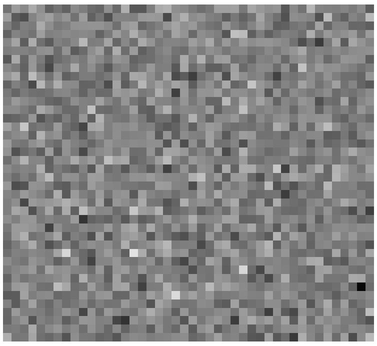


Figure 5.4: Pure white noise grid

An example of such a random grid is given by Figure 5.4 where it has been applied for two dimensions and transformed to greyscale levels.

When neglecting the concept of stationarity, deliberate variability in mean, variance, skewness, kurtosis or even higher moments can be incorporated in the random grid. This can be achieved by making the parameters of the probability distribution a function of position in space, time or space-time. The approach could be implemented by using a continuous function of position or a pre-calculated continuous surface to control one or more of the parameters.

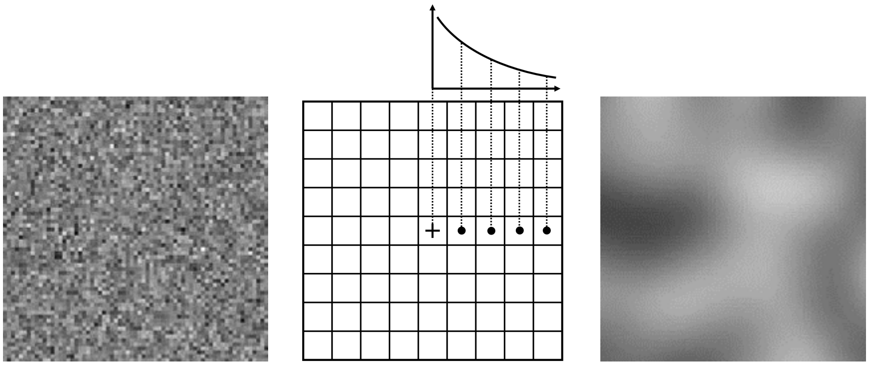
The resulting continuous but inhomogeneous field could then be used to test the capability of the applied interpolation method to cope with such geostatistical anomalies. The present work, however, is limited to the simple case of the parameters mean and variance which remain constant and thus produce a homogeneous model.

Covariance Function Filter The general concept of the theoretical variogram and the associated covariance function has been described in Section 4.3. As already mentioned, the covariance function is used to estimate the parameters of the observed field (see Section 5.3.5), to perform the optimal interpolation (see Section 5.3.6) and for the generation of continuous random grids, which will be described here.

The principle of ceasing correlation, as is immanent to any covariance function, is applied for the moving average filter in order to generate a continuous random field from the pure white noise field. Its application to a two-dimensional field is depicted in Figure 5.5(b). The moving average filter—also called mask, kernel or template for two-dimensional grids [Gonzalez and Woods, 2002]—defines a value for each cell by which the underlying cell (of the grid it is applied to) is to be weighted. The weight is determined by the associated covariance function and the (euclidean) distance of that cell to the centre of the filter.

For practicability, the filter grid has the same dimensionality and resolution as the white noise field grid it is applied to. In the case of a spatio-temporal grid, each particular cell can be specified by its *spatial* (euclidean norm) and its *temporal* distance to the centre. The respective result value given by the associated spatio-temporal covariance function is the weight for that filter cell. Due to the identical dimensionality and resolution, the filter can be applied to a target grid by simple matrix-based translations.

Figure 5.5 shows a continuous random field generated by applying the filter to a white noise field.



(a) Pure white noise array

(b) Moving average filter based on covariance function

(c) Continuous filtered array

Figure 5.5: Random field generation by moving average filter

Depending on the applied covariance function, the filter grid has different extensions. If, for example, a *spherical* covariance function (see Figure 4.4, p. 54) is used for filter definition, the correlation between observations further apart than *range* is always zero. Therefore, the grid size of the filter does not need to extend the corresponding distance for that dimension. Whereas for an *exponential* covariance function where the correlation becomes tiny but never zero, even for large distances, the filter consequently needs to cover all cells of the grid of white noise it is applied to. This means a filter resolution of $2r - 1$, where r is the resolution—for that dimension—of the random field to be generated.

To avoid critical workloads for random grid generation caused by this constellation, an optional restriction is included. The *reach* (not the *range*!) of the covariance function can be restricted to a distance where e.g. less than 1% of full correlation is left. Since the random grid cells affected by these peripheral filter cells are relatively high in amount and relatively low in derived weight, they tend to sum up to zero (relative to the mean value) and can therefore be neglected.

Depending on the size of the filter grid and the current filter position, there is a considerable amount of filter cells that lie outside the target grid. Consequently, they do not contribute to the average value assigned to the target cell on which the filter centre is currently positioned. The proportion of outside filter cells increases towards the fringes and even more towards the corners of the target grid, also depending on the dimensionality.

In some cases, this situation can be avoided by extending the white noise field by $\frac{n-1}{2}$ when n is the resolution of the filter grid in the respective dimension [Oliver, 1995]. This approach was not considered here since no "fringe-effect" of a strikingly different pattern could be identified in the generated fields. Furthermore, it would put considerable burden on the random field generation process, especially for filter grids of large relative extension.

Result: Continuous Random Grid As product of the statistical operation of a moving average covariance-weighted filter on a pure white noise grid, the continuous random grid has properties that are determined by this process. Since the process of filtering basically generates weighted mean values of the surrounding random cells, the derived filtered grids—at least in tendency—

share their mean value with the white noise field used to generate them. From the configuration of the variogram based filter (Figure 5.5(b)), also the variance of each cell value of the filtered grid can be derived as variance of the weighted sample mean with

$$\sigma_{\overline{x}}^2 = \sigma_0^2 \sum_{i=1}^{n} w_i^2, \tag{5.1}$$

where σ_0^2 is the variance of white noise field that is equal for each cell and the w_i are the weight values derived from the covariance function for each position in the filter grid.

The value of $\sigma_{\overline{x}}^2$—as being derived from the white noise field and the filter grid configuration—determines the variance of each single cell of the random grid. This value is assumed as approximate dispersion variance and is therefore used as initial value for the variogram fitting procedure (see Section 5.3.5), although it is not to be confused with the *sill variance* in the strict sense [Webster and Oliver, 2007, p. 102].

5.3.2 Sampling

The sampling design has to be sufficient with respect to density and distribution to capture the underlying phenomenon in a way that adequately addresses the problem or question at hand [Chun and Griffith, 2013]. Some general issues about the effectiveness and efficiency of sampling have already been mentioned in Section 3.3.2. These considerations will be concretised in the following.

Within a monitoring scenario, sampling is the most fundamental and often also the most expensive task; all subsequent process steps must rely on this limited data about the real phenomenon that is provided by sampling. There are the following aspects that have to be considered carefully in this context:

- the phenomenon itself and its properties (dynamism in space and time, periodicities, isotropy and trends)
- the sampling design (density and distribution, effectiveness and efficiency of observations)
- the sensor accuracy
- the appropriateness of the selected interpolation method
- the problem to solve or the question to answer

When regarding a monitoring scenario as a whole, it turns out that these aspects are not independent but relate to each other, as following examples show:

- An increased dynamism of the phenomenon makes a higher sampling rate necessary
- To choose an appropriate and elaborate interpolation method can help to reduce the number of necessary observations
- The more complex a process is, the more observations are usually necessary to generate a model that adequately represents its dynamism
- A dense network of observations is necessary to gain sophisticated knowledge about a phenomenon and thus helps to refine the associated models
- The better the physical processes are understood, the less observations will eventually be needed to generate an appropriate model
- Changed requirements according to the aims of the monitoring—e.g. detailed reconstruction instead of rough aggregation—will probably affect the overall effort that is necessary for the monitoring

In many cases, the dynamism of the continuous field is only roughly known in advance and is therefore not revealed until processing the data. In the case of geostatistics, the experimental (see Section 4.2) and the theoretical (Section 4.3) variogram derived from the data will contain hints whether the chosen model is appropriate. It is then up to the operator to decide if the sampling and/or the interpolation model need to be improved. The monitoring framework might provide suitable indicators—e.g. residuals from the variogram fitting—to support such decision processes.

Beside the "reconstruction" of a continuous phenomenon in space and time—

e.g. as a representation of its current state—monitoring can also include the task to check the model for predefined critical states and fire an alert when such a state is present (see Figure 5.13, p. 100, also [Lorkowski and Brinkhoff, 2015a]).

Such a critical state could be defined by an exceeded threshold in the simple case. For more elaborate applications, it might be formulated as follows: “We need to make sure by 95% confidence that the nitrogen oxide pollution of district A is below $40\mu g/m^3$ in average per day.” Therefore, it is not sufficient to rely on the interpolated values alone; their confidence estimation also needs to be considered here. Only the combination of value and confidence estimation will provide enough information to either confirm or reject the presence of a critical state, which, by this particular definition, could also be caused by insufficient sampling. Since the method of kriging explicitly comprises confidence estimation, for that reason alone it is an appropriate solution for problems similar to the one described above.

Sampling Density

As already mentioned above, the sampling density that is necessary for an appropriate monitoring of continuous fields depends on its dynamism in each dimension. In practice, this dynamism is either known from previous or similar monitoring scenarios or has to be derived directly from the data (see sections 4.2, 4.3, 6.2).

The issue is identified as “data sufficiency problem” in [Sun and Sun, 2015, p. 22], which should be addressed by an “optimal experimental design (OED)”. It strives for a good compromise between information content and cost. This problem is referred to as “representativeness” by [Meyers, 1997, p. 187]. Consequently, in the realm of monitoring continuous phenomena, some estimation of when a sampling density is *sufficient* is necessary.

Within the proposed simulation framework, the dynamism can be determined by setting the spatial, temporal or spatio-temporal parameter(s) *range* of the variogram filter used to generate the reference field (see Section 5.3.1). These parameters can then be compared with the ones derived from the variogram fitting procedure applied to the simulated observations (see Section 6.2).

The main objective that is addressed here is to quantify the relation between this dynamism and the average sampling density that is necessary to capture the phenomenon adequately. Instead of applying heuristics like "nested survey" [Webster and Oliver, 2007, p. 127] in order to systematically inspect the autocorrelation structure of a particular phenomenon, we will try to find some law, or at least some rule of thumb, to derive the necessary average sample rate from the extension and the dynamism of the phenomenon. If this rule is valid for synthetic fields, it is assumed to be applicable to real-world phenomena for which the dynamism is estimated by the parameter *range* for each dimension.

To approach this problem, it is first reduced to the one-dimensional case before looking for analogies to the Nyquist-Shannon sampling theorem which is well known in signal processing [Pollock et al., 1999].

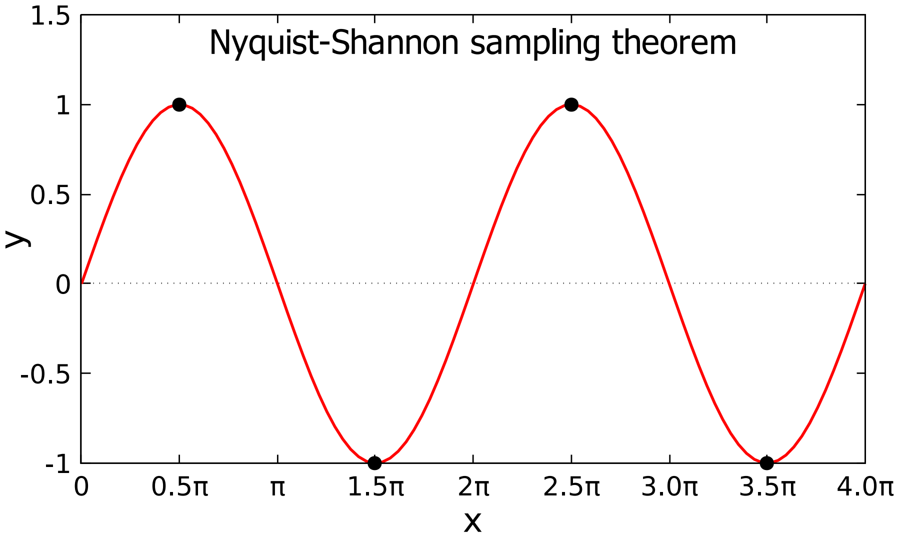


Figure 5.6: Nyquist-Shannon sampling theorem

As can be seen in Figure 5.6, according to the theorem, at least two samples per wavelength are necessary to capture a periodic sine signal. The sampling distance d is thus determined by

$$d = \frac{\lambda}{2}. \tag{5.2}$$

Since periodicity is usually not found in natural continuous fields, the sample rate necessarily needs to be higher to capture such phenomenoa appropriately. For approximation, uniformly randomly distributed samples instead of systematic or stratified sampling [Chun and Griffith, 2013] are presumed, because such regular configurations are hardly encountered for mobile wireless sensors [Umer et al., 2009]. Furthermore, systematic or stratified distributions will reduce the variety of small distances which are crucial for variogram estimation [Oliver and Webster, 2015, p. 53], [Armstrong, 1998, p. 53].

Two general challenges need to be addressed to transfer the principle of Nyquist-Shannon to the domain of multidimensional continuous phenomena:

1. Finding a reasonable factor to relate the wavelength of a periodic signal to the parameter *range* of general continuous phenomena
2. Extending the principle from one-dimensional to multidimensional applications

In order to obtain an estimate of the geostatistical concept of *range* within a sine signal, the experimental variogram for 100 uniformly distributed observations within one wavelength of a sine function is generated. From this, the semi-variance can be derived by Equation 4.2 for each pair of observations. The experimental variogram is generated by plotting these values against their corresponding pair distance (see Section 4.2). As can be seen in Figure 5.7, the experimental variogram for the 4950 possible pairings from 100 uniformly dispersed observations converges against zero as the distance approaches the value of 1 (unit: wavelength of 2π) and is confined by sine shaped upper and lower bounds. These regular patterns are caused by the periodicity and are usually not found in experimental variograms. But this can be neglected here since we are only interested in the *first* position from which on the dispersion of values exceeds the total variance.

As common in geostatistics, the trend of the variogram can be approximated by interval-wise aggregations of the experimental variogram points [Webster and Oliver, 2007, Gräler et al., 2016]. The polygon connecting those aggregation points represents this trend. It is this geometry to which a theoretical variogram is usually fitted by adjusting its parameters (and therefore the parameters of the covariance function, see Section 5.3.5).

In the case of the sine signal, however, there is no appropriate theoretical variogram to fit to since the semi-variances γ decrease when approaching the value of 1.0, which is not the case for a valid variogram. But since the only

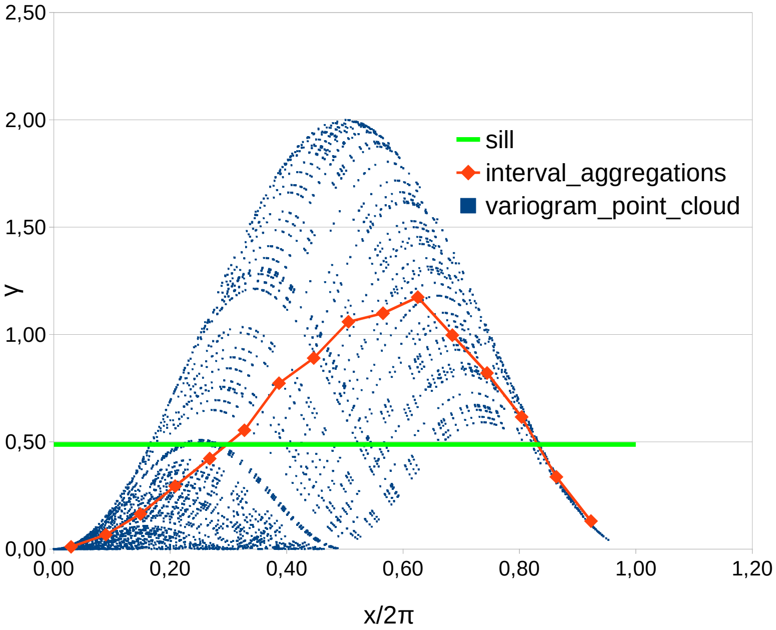


Figure 5.7: Experimental variogram with semivariances (γ) plotted against pair distances (normalized to the wavelength of 2π) of observations on the sine signal; the *sill*, represented by the green horizontal line, is intersected by the polygon of aggregated interval points; the abscissa position of this intersection is considered as *range*

value of interest is *range* here, there is a quite straightforward way to roughly estimate it.

As can be seen in Figure 5.7, an approximation of the value *range* can be derived from the first point of intersection between the total variance (or dispersion variance [Webster and Oliver, 2007]) of the dataset (horizontal line) with the polygon line.

This point is assumed to represent the threshold distance between point pairs from which on the dispersion (or semi-variance) between the point values is just as large as the total variance of the dataset. As already mentioned in Section 4.3, this does not strictly comply with geostatistical practice, but is considered to be sufficient to derive an approximative value for the minimum sampling density.

The experiment as depicted in Figure 5.7 was repeated 30 times and in average reveals a dispersion (or total) variance of 0.4878 with standard deviation of 0.0503. The average ratio between wavelength λ and range r is 0.2940 with standard deviation of 0.0188. For convenience, this value is rounded to the safe side (down), so the range-wavelength ratio is estimated by

$$\frac{r}{\lambda} \approx \frac{1}{4}. \tag{5.3}$$

To adequately capture a sine-shaped signal for interpolation by the method of kriging, we assume an minimum average coverage by two observations per range distance (or eight observations per sine wavelength, respectively) because this is the minimum number of observations to at least *detect* a correlation above the average correlation within one range distance. From that, the sampling distance d_p can be derived by

$$d_p = \frac{\lambda}{8} \tag{5.4}$$

to capture *periodic* signals of wavelength λ for kriging interpolation and

$$d_c = \frac{r}{2} \tag{5.5}$$

for *continuous* non-periodic signals with range r.

To derive the number of samples for regions of arbitrary extension, we need to apply a factor f that represents how many times the (average) sampling distance d_c is contained within the extent e:

$$f = \frac{e}{d_c} \tag{5.6}$$

Together with Equation (5.5), we can now determine the number of samples c necessary per dimension i by

$$c_i = 2\frac{e_i}{r_i}. \tag{5.7}$$

The fundamental relationship between a continuous phenomenon of range r, the extent e and the number of observations c_1 necessary to capture it, is thus defined. It can be used to estimate sampling density for *one* dimension. To generalize the concept in order to be applicable for multiple dimensions, we calculate the product of its n dimension-wise representatives by

$$c_n = \prod_{i=1}^{n} 2\frac{e_i}{r_i}. \tag{5.8}$$

This expression is used to estimate the minimum number of uniformly distributed random samples on multidimensional continuous fields as a function of their values for extent and range for each dimension. Thus, we can determine an appropriate sampling density for arbitrary initial configurations of *sill* and *range* in the random reference model. The approach is experimentally validated in Section 6.1.

Result: Multidimensional Point Set Depending on the simulated or actual process of sampling, the set of observations represents the degree of knowledge about the observed phenomenon that is available. The geostatistical properties (mainly the parameters *sill* and *range*) of the theoretical variogram are not necessarily equal to the ones within the observed region, therefore it is also called the *regional variogram* [Webster and Oliver, 2007]. In practice however, the properties derived from the observations are often the only ones available and thus have to be worked with.

For a good estimation of the geostatistical properties of a field, the distribution of observations should sufficiently cover all distances relevant for the given problem to provide enough information for variogram fitting [Chun and Griffith, 2013, Webster and Oliver, 2007]. Especially the short distance are of decisive importance for variogram estimation [Armstrong, 1998, p. 53], [Oliver and Webster, 2015, p. 53].

With real world data, there might be anomalies like anisotropy that will usually materialize in the derived experimental variogram cloud. If no other geostatistical property information is available (e.g. from previous samples), the set of observation points is carrier of both (i) the discrete spots of knowledge about the phenomenon to be interpolated between and (ii) the statistical properties this interpolation has to be based on [Wackernagel and Schmitt, 2001].

5.3.3 Experimental Variogram Generation

The experimental variogram has already been introduced as basic geostatistical concept in Section 4.2 and was also applied to determine the *range* property of a sine-shaped signal in Section 5.3.2.

Given a sufficient number of observations as described in the previous section, we will now set out the process steps that are necessary to derive the parameters of the theoretical variogram from them. Namely, these steps are (i) the generation of the experimental variogram (this section), (ii) the aggregation of the variogram points (Section 5.3.4) and (iii) the variogram fitting (Section 5.3.5).

For better visual demonstration of the method, we stick to a two-dimensional continuous random field generated with following parameters:

- white noise field: 150 x 150 grid cells, $mean = 5000$, $deviation = 500$
- variogram filter: separable gaussian, range of 75 grid cells, resulting grid cell value deviation of 4.65
- sampling: 20 points (derived by Equation 5.8), uniformly distributed
- experimental variogram: 190 variogram points comprising of spatial distance and semi-variance γ derived from the sample point pairings
- aggregation: 16 aggregates (by Equation 5.10 with $b = 1.5$, $c = 0.8$), partitioning dimensions aggr. method: median

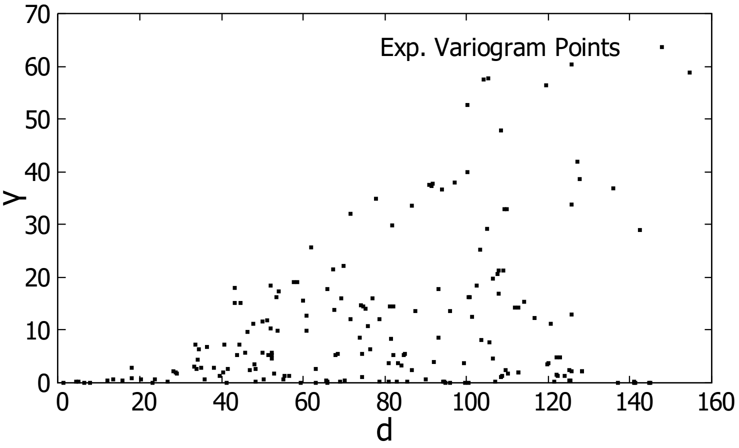


Figure 5.8: Experimental variogram point cloud

As can be seen in Figure 5.8, the semivariances (ordinate) tend to scatter more with increasing pair distance (abscissa). A spatio-*temporal* variogram [Cressie and Wikle, 2011, Gräler et al., 2016] can be visualized as three-dimensional plot with spatial distance, temporal distance and semi-variance as axes (see Figure 4.1).

Although the number of variogram points would also allow for direct fitting of the theoretical variogram in this example, we will apply binary space partitioning (BSP) as aggregation approach, since its principle is more comprehensible with small datasets. It will be introduced in the next section.

5.3.4 Experimental Variogram Aggregation

Before the interpolation by the geostatistical method of kriging can actually be carried out, a formal description of the spatio-temporal autocorrelation is needed. After generating the experimental variogram as a point cloud in the previous step, the theoretical variogram function associated with this covariance function (see Section 4.3) has to be fitted to this point cloud [Müller, 1999, Brunell, 1992, Gräler et al., 2012]. The number of variogram points in the experimental variogram n_v depends on the number of samples n_s by

$$n_v = \frac{n_s^2 - n_s}{2}. \tag{5.9}$$

The subsequent and rather complex step of variogram fitting can therefore become too expensive for large datasets. The common solution for this problem is to perform some aggregation that retains the general dispersion characteristics of the original variogram points [Webster and Oliver, 2007].

Depending on the dimensionality of the observational data and the dimensionality of the associated variogram model (spatial/temporal/spatio-temporal, isotrop/anisotrop), the aggregation of points in the respective experimental variogram will have to be possible with various dimensionalities to be generic.

The common structure of the experimental variogram for all dimensionalities is that of n independent variables (e.g. spatial and temporal distances between pairs of observations) and one dependent variable, which is the semivariance γ as given by Equation 4.2.

This is also the target variable of the theoretical variogram to be aggregated by using the spatio-temporal proximity of points as criterion for grouping. Therefore, the common concept of aggregating the original variogram points by using intervals of constant lag intervals [Webster and Oliver, 2007] is extended (or generalized) by the following features:

- Instead of using only one dimension for segmentation of the experimental variogram, binary space partitioning (BSP, [Samet, 2006]) allows for multidimensional segmentation
- Instead of rigid segmentation (e.g. by constant interval size), more flexibility is achieved by variable hyperplanes that can adapt to the given data structure

This approach aims at a generic and robust solution for the central problem of geostatistics: variogram fitting. It provides flexibility in terms of

- the dimensionality of the used variogram model
- the number of points of the experimental variogram
- the dispersion of points

As already mentioned, the experimental variogram is a set of points in $\mathbb{R}^d$, where $d-1$ dimensions represent parameters of the particular variogram model and one dimension represents the target variable γ. For aggregation, all dimensions except that of γ are used for binary space partitioning, thus generating disjunct regions with subsets of the original variogram point set.

According to its common purpose of space partitioning for search operations, a BSP (binary space partitioning) tree usually—e.g. when implemented as k-d tree—subdivides a set of objects into two subsets of equal number of elements. In a recursive manner, the partitioning dimensions or axes do cycle according to a predefined order [Samet, 2006].

For the sake of adaptivity, the partitioning method is extended to let diverse statistical parameters control the process. Figure 5.9 illustrates critical decisions within the BSP tree algorithm where statistical properties are used to determine how the partitioning is carried out. Therefore, the algorithm keeps track of the statistical properties of each dimension separately. So for each set of points the minimum, maximum, extent, mean, median and variance are calculated per dimension and can thus be used as control parameters for the points of decision that are described below.

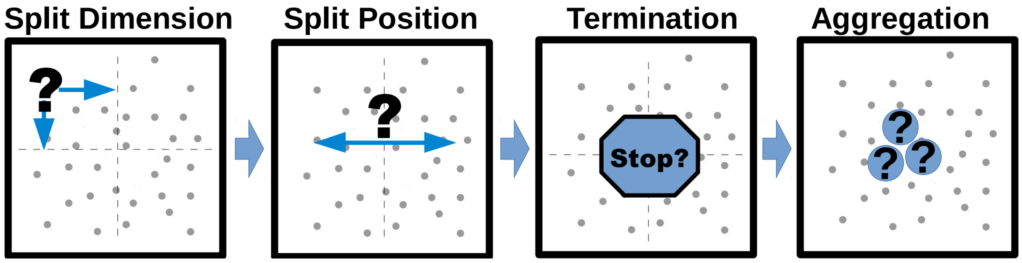


Figure 5.9: BSP tree partitioning process: options for control by statistical properties

Split Dimension If there is more than one free variable in the variogram, as is the case for spatio-temporal models, the recursive partitioning algorithm needs to select the next splitting dimension, or, in other words, which coordinate axis will be the normal vector of the next splitting hyperplane.

The statistical properties described above can be used to determine the next dimension. So it might be reasonable to select the dimension with the greater extent or deviation for the next split. In many cases it is appropriate to relate this value to the one of the total set that the procedure started with to get a relative value. This variant was also applied here.

For a uniform splitting pattern, the algorithm can also simply toggle between all dimensions without any parameter checking. This is the behaviour of a standard k-d tree [Samet, 2006].

Within this solution, consecutive splits by the same dimension are allowed, which might be useful for very anisotropic point distributions. More complex definitions where several parameters and conditions are combined might also be reasonable to adapt to such data structures, but they are not regarded here.

Split Position Once the dimension for the next split is determined, the position of the hyperplane on the corresponding axis has to be set. The statistical properties of this particular dimension can be used for the determination of this position.

In the case of the median value as split position, subsets with equal numbers of elements are obtained, which is the case for k-d trees [Samet, 2006]. Alternatively, the mean value can be used to give outliers more influence than with the robust median. Simply selecting the middle position will result in a regular grid, but only when split dimensions toggle and tree depth is equal for each dimension. All of these variants are tested and evaluated in Section 6.2 (see Table 6.1, p. 142).

Termination There are several conditions that can be used within a BSP tree algorithm to terminate the recursive partitioning. In the context of aggregation of the experimental variogram, the following are considered reasonable:

- maximum tree depth
- maximum elements per leaf
- maximum total number of leaves
- spatial extent of current leaf

Each of these variants has its advantages and drawbacks: A constant maximum tree depth is easy to implement, but does eventually not adapt well to the given data structure. A constant maximum spatial extent of leaves is also straightforward, but may produce subsets with numbers of elements differing too much. A constant total maximum number of leaves is difficult to implement in a recursive manner if the tree should not become too unbalanced.

In order to achieve robust behaviour, more complex termination rules can be defined by the logical combination of multiple conditions.

For this study, the termination condition of maximum elements per leaf was implemented. From the algorithmic perspective, this condition is fulfilled when stopping the recursive partitioning as soon as the threshold number of elements is achieved or undercut. The condition produces statistically similar subsets of points to be aggregated. For reasonable sizes of those subsets while given arbitrary total amounts of elements, the logarithm-based formula

$$n_a = c \cdot log_b(n_t) \tag{5.10}$$

is used, where n_a is the total number of aggregated points to be created, n_t is the number of points in the original variogram, b is a logarithmic base that controls the degree of decreasing, and c is a linear scaling factor. By applying this formula, an arbitrary choice of the number of aggregations is avoided. It adapts to the total amount of original variogram points by producing reasonable and feasible numbers of aggregated points.

Aggregation The preceding procedure provides n datasets of the original experimental variogram dataset, separated by BSP hyperplanes. To actually *aggregate* these sets to one point for each of them, there are several options taken into account:

- mean value
- median value
- middle of the corresponding BSP tree partition interval

These options can be assigned individually to each of the independent dimensions used for BSP tree partitioning. For the target variable γ itself, only the mean value is assumed to aggregate the dispersion correctly [Oliver and Webster, 2015, p. 16], [Cressie, 1993, p. 59]. These variants are also tested in Section 6.2 (see Table 6.1, p. 142).

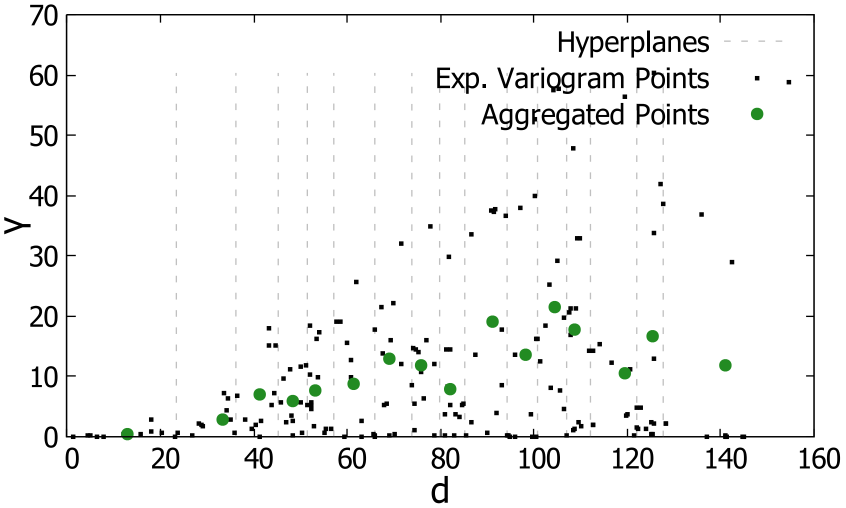


Figure 5.10: Aggregation of variogram points; different interval sizes result from adaptive BSP algorithm

Figure 5.10 illustrates the BSP aggregation that is applied to the point cloud from Figure 5.8. Since the points of the experimental variogram represent the statistical properties of the dataset, the aggregation is supposed to be carried out in a way that transmits, at least approximately, the significant properties of the original point cloud to the reduced point set.

As can also be seen from the plot, the aggregated set of points is by far less dispersed than the original variogram point cloud and already indicates a continuous function. Beside the geometrical properties, each aggregation produces additional statistical data like variance or skewness that could be used to define weights for the subsequent process step of variogram fitting [Gräler et al., 2016]. But since the aggregations are already statistically similar due to the termination condition of maximum elements per leaf, this mechanism is not considered here.

5.3.5 Variogram Fitting

The aggregation of the original experimental variogram generates a dataset of reasonable size for the fitting of the parameters of a theoretical variogram. Because of the redundancy of data points, they will not fit the theoretical variogram and a non-linear optimization method like Gauss-Newton has to be used to estimate its parameters [Sun and Sun, 2015, Aigner and Jüttler, 2009].

As already mentioned in Section 4.3, the theoretical variogram is the mirror image of the covariance function [Webster and Oliver, 2007, p. 55]. So by fitting the theoretical variogram to the points that were aggregated from the experimental variogram, we obtain the parameters of the respective covariance function needed for kriging.

Generally, the problem can be defined by fitting the variogram model

$$\gamma = f_p(x) \tag{5.11}$$

with x being the distance (spatial, temporal or spatio-temporal) for which the variogram γ is returned.

The Gauss-Newton algorithm [Sun and Sun, 2015, van den Bos, 2007] iteratively determines the vector of parameters $p_1, ..., p_k$ that minimize the squared residuals between the observations (here: the aggregated variogram points) and the function values at the respective positions [Wang et al., 2006, Aigner and Jüttler, 2009]. By equipping each data point with a weight w_i, the process can consider specific circumstances that are supposed to have influence on the optimization result.

The distance to the origin of the variogram, the number of points used for the preceding aggregation or the variance of their mean value are reasonable candidates to define the weights [Cressie, 1985]. Since approximate equal number of points per aggregate are provided by the BSP algorithm, the weighting is derived from the (n-dimensional) distances to the origin.

Weighting

In order to get a good estimate of the variogram at its origin, the points near the ordinate should be weighted stronger than the more distant ones [Armstrong, 1998, p. 53], [Oliver and Webster, 2015, p. 53]. A weighted variant of the Gauss-Newton algorithm was implemented to achieve that higher influence of the elements of low n-dimensional distance by defining the weights per aggregated point by

$$w_j = \prod_{i=1}^{n} 1 - \frac{d_{ji}}{max(d_i)}, \tag{5.12}$$

where d_{ji} is the distance of the aggregated point j to the origin regarding dimension i and $max(d_i)$ is the maximum of that distance that occurs in the whole set of aggregated points. For each dimension and therefore also for the product, the value is guaranteed to be between 0 and 1.

A smoother decrease of weight by distance is achieved by the sine-based function:

$$w_j = \prod_{i=1}^{n} 1 - sin^2(\frac{\pi d_{ji}}{2}). \tag{5.13}$$

Alternatively, to achieve a stronger differentiation of weighting between points near to and points far from the coordinate origin, a weighting function based on a variant of the logistic function [Rasmussen, 2006] was applied with

$$w_j = \prod_{i=1}^{n} 1 - 1/(1 + exp(g(1 - 2\frac{d_{ji}}{max(d_i)}))), \tag{5.14}$$

where factor g controls the gradient. It is set to 5.0 in Figure 5.11 which contains plots of the functions for one dimension. The weighting variants introduced here will be evaluated experimentally in Chapter 6.

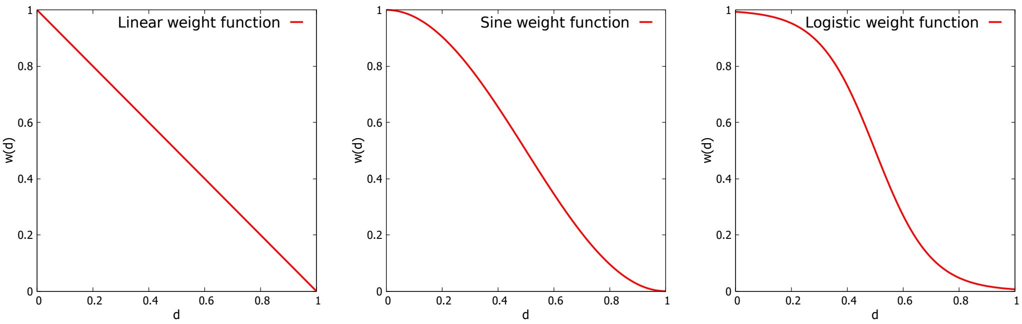


Figure 5.11: Weighting functions relating relative distance to origin (d) to weight (w): (1) linear, (2) sine-based and (3) logistic

The number of points, the variance, or other parameters from the preceding aggregation process could be used alternatively or be incorporated into the proposed weighting functions to give them some influence on the Gauss-Newton procedure.

This variant was not implemented in the framework because it is assumed that the combination of adaptive aggregation and distance weighting is sufficient to cover this aspect for the present study. But this or similar variants could easily be evaluated using the tools for systematic variation and evaluation (see Section 5.5).

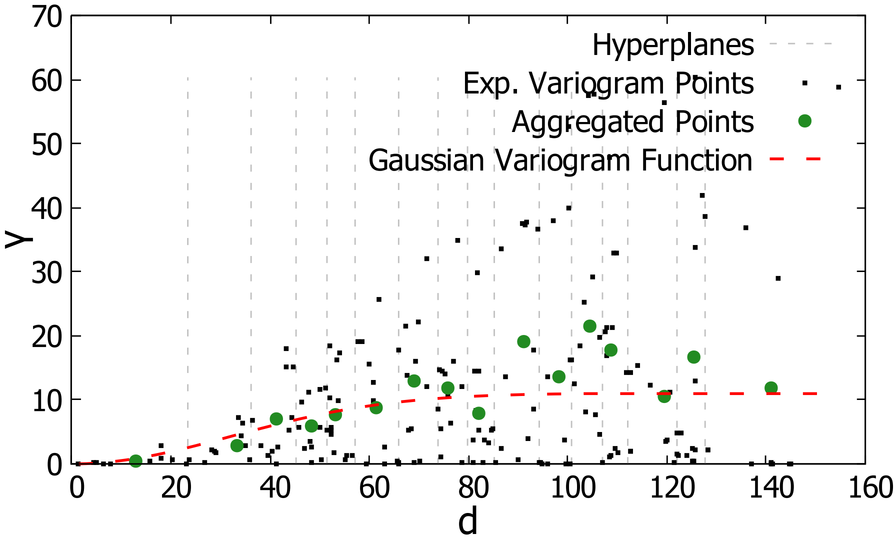


Figure 5.12: Theoretical variogram fitted to aggregated variogram cloud

Figure 5.12 shows a Gaussian variogram model fitted by linear weighted aggregation points. The parameters that were illustrated in Figure 5.9 are set as follows: split dimension: not applicable here, since variogram contains only one spatial dimension; split position: median; termination: maximum point number; aggregation: mean. The logistic function was used for the weighting of the subsequent fitting by the Gauss-Newton algorithm.

As the graph reveals, the algorithm yields a reasonable fitting to the variogram points at sight. The *visual* assessment is still an important issue of variogram fitting [Oliver and Webster, 2015, p. 38 f.], [Armstrong, 1998, p. 54]. There are also approaches to completely automatize the fitting [Pesquer et al., 2011, Desassis and Renard, 2013]. But regarding the sheer amount of varieties of kriging and its associated parameters, the selection and fitting of the variogram is therefore a very complex task for which no established solution is available yet (see Chapter 7).

Multiple Initial Values: Hybrid Approach

As is a common problem for non-linear optimization algorithms, also the Gauss-Newton method is not guaranteed to converge for every constellation of initial values [Sun and Sun, 2015, p. 63]. The aggregation of the variogram points and the weighting scheme already reduces this risk, but does not completely exclude it.

To address this problem and to get better results in case of multiple local minima, the optimization procedure is started with varying initial parameter values. The variants are generated by n-dimensional subdivision of the parameter value or values. Thus, a set of starting parameter variants of size $x \cdot n$ is generated, where x is the number of subdivisions per dimension and n is the number of dimensions of free parameters.

It is not the whole value domain that is used as initial interval to be subdivided per dimension. Instead, the thresholds are determined by robust estimation based on quantiles.

Given this set of initial parameter settings with predefined criteria when the iteration shall cease—for both cases: sufficient converging as well as diverging behaviour—leads to a set of result values with usually different values of residuals.

In the ideal case, all starting parameter variants converge to the same result, which is only the case for very robust constellations. Except for ill-conditioned constellations of sampling, this approach provides a robust estimate of the parameters by selection the iteration solution with minimum residuals. Local minima are more likely to be found this way.

Depending on the sensitivity of the given constellation, the amount of initial values can be of decisive importance for achieving an optimal solution. With too many variants, however, this complex process might exceed the computational capacities.

In the experiments carried out in Section 6.2, a moderate amount of variants was sufficient to yield feasible solutions. For situations where this is not the case, more sophisticated methods to improve convergence should be applied [Andradóttir, 1998, Sun and Sun, 2015, Schittkowski, 2002].

5.3.6 Kriging

Once the appropriate variogram model is determined, the interpolation procedure itself includes the inversion of the covariance matrix (once per model) and its application to determine optimal weights by which each observation contributes to the estimated value (once per interpolation). Based on these weights, kriging also provides a confidence estimation for each interpolated point. The general proceeding of kriging has already been set out in Chapter 4.

With big numbers of observations, the inversion of the covariance matrix might produce critical workloads due to its complexity of $\mathcal{O}(n^3)$ [Gelman et al., 2014, p. 503], [Sun and Sun, 2015, p. 356]. There are various approaches that address this issue [Wei et al., 2015, Henneböhl et al., 2011, Pesquer et al., 2011, Cornford et al., 2005, Barillec et al., 2011, Osborne et al., 2008].

In this work, an approach is introduced that addresses the problem of computational burden *and* the problem of continuous integration of new observations into an existing model (see Section 5.4.2). It exploits the estimation variance (kriging variance) that is provided by no other interpolation method [Oliver and Webster, 2015, p. 1].

5.3.7 Error Assessment

Given the same extent and resolution for both the continuous reference random field and the one derived by interpolation of observations, the deviation between those two models can easily be calculated. The RMSE provides a compact indicator for the overall quality of observation and interpolation. The effect of changed method variants or parameters (see Section 5.5) can thus be quantified.

An error map or map of the "second kind" [Meyers, 1997, p. 464] of the same resolution can help to reveal more subtle patterns indicating systematic flaws of the monitoring process (see Section 6.6).

Representing a single-number summary of the error map, the indicator is given by

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}}, \tag{5.15}$$

where for each grid cell i, $\hat{y}_i$ is the value of the derived model, y_i is the value of the reference model and n is the total number of grid cells of the model.

Unlike the situation in a real world monitoring scenario where interpolation quality has to be estimated by approaches like cross validation [Gama and Pedersen, 2007, p. 147], the synthetic reference model of arbitrary resolution allows total transparency of the errors caused by sampling and interpolation. While the RMSE will in most cases be sufficient to compare the interpolation quality of different monitoring process variants, the more verbose representation of the error as deviation map can provide valuable hints for further improvement.

Geometric patterns within the deviation map that significantly differ from pure random fields can indicate potential for systematic improvement of the monitoring process. So different patterns in the error map might provide visual hints to particular deficits in the monitoring process that produced the associated model.

- A predominantly high error value that is only mitigated regionally at the spots around the observations might indicate an insufficient density of samples
- Distinctive border areas of high slope (discontinuities) that separate regions of rather homogeneous error values are a hint for an insufficiently fitted variogram model
- A rather continuous error map with moderate error values at spots of maximum isolation from observations indicates a near optimal configuration of the monitoring

While the RMSE provides a straightforward quantification of model quality that can be used to easily identify the best among many solutions, the error map reveals subtle patterns that might contribute to more thorough investigations.

Both approaches, however, do only work with synthetic models or with phenomena that are available at much finer resolution than necessary in the planned monitoring scenario in order to serve as reference for experimental study. In this work, error assessment is applied to both synthetic phenomenon models (see Section 6.2) and real remote sensing data (see Section 6.6).

5.4 Performance Improvements for Data Stream Management

Monitoring of continuous phenomena poses several specific challenges according to the processing and the archiving of observations. Some of them that are considered to be crucial are addressed in this section.

So providing an interpolated grid from a set of discrete observations means considerable computational burden if massive data or real-time requirements (or both!) is present. Also, the seamless and efficient actualization of a calculated model by new incoming observations is indispensable for (near) real-time monitoring systems. Both problems are addressed by the approach that is set out in Section 5.4.2.

Although storage costs are continuously decreasing, the archiving of extensive observational data might nevertheless reach critical dimensions. Whilst grid data as derived from interpolation provide better interoperability, retaining the original observational (vector) data has several advantages (see Section 3.2.2). A compression algorithm specifically designed for such data is introduced in Section 5.4.3.

5.4.1 Problem Context

The specific features introduced in the next two sections can best be considered in the context of a monitoring system architecture as sketched in Figure 5.13.

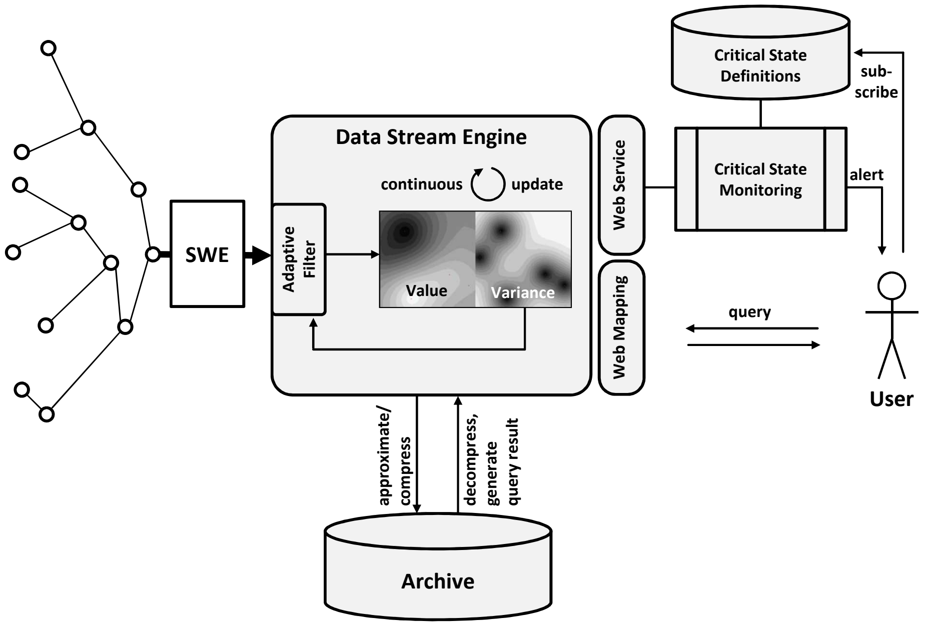


Figure 5.13: Architecture of a system that processes, visualises, monitors critical states, and archives sensor data streams

The envisioned data stream engine (DSE) [Gama and Gaber, 2007] continuously processes incoming observations (provided by sensor web enablement (SWE)) and integrates them into the model that reflects the current state, eventually as Web Map Service (WMS) [Blower et al., 2013]. Beside the value of interest, the model also keeps track of the deviation map as "map of the second kind" [Meyers, 1997, p. 464].

As already mentioned in Section 4.5, this deviation map or variance map can be used for several purposes. It can indicate insufficient confidence for the critical state monitoring. For massive loads of data, it can be used as adaptive filter to only let non-redundant observations pass. Its role as weighting schema for the merging of sub-models—in order to mitigate computational workload or to support a continuous update by new observations—is subject of the next section.

From the user's perspective, the monitoring system should provide the model at arbitrary points in space and time. So the data prior to the current model need to be archived and retrieved appropriately. As indicated by Figure 5.13, the process of compression and decompression ought to be hidden from the user who usually accesses the data by some (web) interface.

Based on web services, a data stream engine (DSE) should provide interfaces for both interactive web mapping and automated monitoring. For the latter, critical states can be defined, subscribed to a specific database and checked against the current map regularly. Such definitions can refer to values (e.g. for an alert after an exceeded threshold), confidence estimations (when more measurements are necessary) or both combined (high risk of exceeded threshold [Guttorp, 2001, p. 24]).

For queries on historical data and for long-term analyses, an archive containing data that are compressed by approximation is maintained alongside with the real-time services. When queried, it is decompressed and provided as usual map.

The methodologies introduced in the next sections are in principle designed to support the functionalities of an environment as sketched above.

5.4.2 Sequential Model Merging Approach

Overview

As already mentioned, the sequential merging approach addresses two common problems in the context of monitoring continuous phenomena:

1. Reducing the computational workload for big datasets

2. Allowing for subsequent and smooth model updates for data stream environments

The general task that these problems are associated with is to generate a regular grid from (potentially) arbitrarily distributed and asynchronously conducted discrete observations. The general interpolation problem is a subject matter of spatio-temporal statistics [Cressie and Wikle, 2011], while the peformance issue is often also addressed by data stream management [Gama and Gaber, 2007]. Consequently, the proposed solutions for these problems depend much on the context they are tackled from.

Related Work

A new design of a data stream engine (DSE) that is based on k Nearest Neighbors (kNN) and spatio-temporal inverse distance weighting (IDW) is suggested by [Whittier et al., 2013].

It uses main memory indexing techniques to address the problem of real-time monitoring of massive sensor measurements. In contrast to this approach, we want to avoid a sub-model based on a fixed sized temporal interval. By merging sub-models continuously, we also consider old observations if no better information is available. This might be especially important when observations are inhomogeneously distributed in space and time.

Trend clusters in data streams are discussed as techniques to summarize, interpolate and survey environmental sensor data by [Appice et al., 2014]. Since one main application is the detection of outliers within a rather low dynamic phenomenon (solar radiation), the approach allows a coarse approximation by clusters of similar values. For our purpose, a smooth representation of each state is desirable.

In [Walkowski, 2010] the kriging variance is used to estimate a future information deficit. In a simulated chemical disaster scenario, mobile geosensors are placed in a way that optimises the prediction of the pollutant distribution. Instead of optimising the observation procedure itself, we exploit the kriging variance in order to achieve efficient continuous model generation from massive and inhomogeneous data.

The decomposition of a spatial process into a large-scale trend and a small-scale variation is carried out in [Katzfuss and Cressie, 2011] to cope with about a million of observations. This solution is an option for optimizing very large models, but is not helpful for our sequential approach with its real-time specific demands.

A complex model of a *Gaussian process* (synonym for kriging) that incorporates many factors like periodicity, measurement noise, delays and even sensor failures is introduced by [Osborne et al., 2012]. Similar to this work, sequential updates and the exploitation of previous calculations are performed, but on a matrix algebra basis. It uses kriging with complex covariance functions to model periodicity, delay, noise and drifts, but does not consider moving sensors.

Requirements

Concluding from the state of the problem area as characterized by the related work above, the following requirements are considered to be crucial within the scope of this work:

- Locally confined, smooth and flexible updates of interpolated models
- Preserving confidence estimate (kriging variance) as crucial information also for adaptive filtering and critical state checks (see Section 4.5)
- Provision of immediate coarse results generated by subsets of observations
- Preserving of preceding computational effort

Principle

The sequential merging approach that is set out here exploits the variance map provided by kriging using Equation 4.11. Depending on the purpose, it can also be represented as *deviation* map (see Figure 5.14).

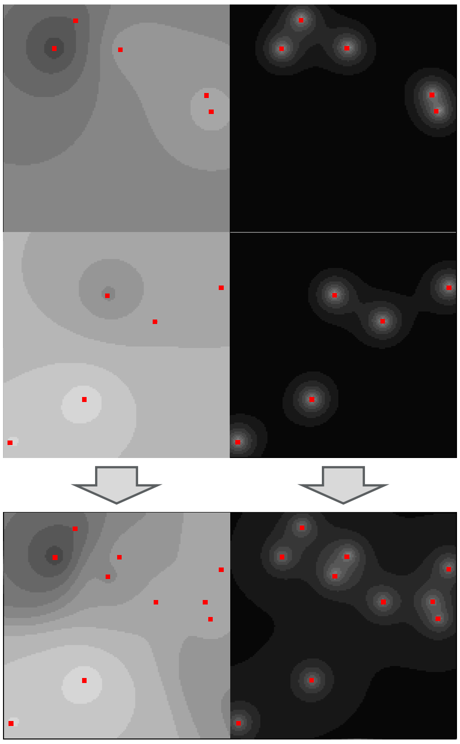


Figure 5.15: Merging of models by using weight maps: the values (l) and variances (r) of two models are merged to a resulting model that combines the information they contain (bottom)

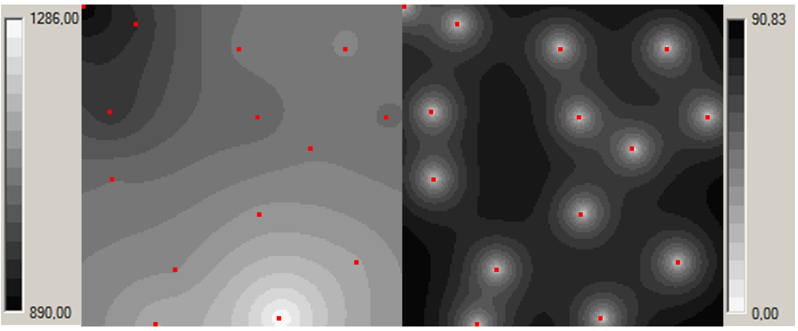


Figure 5.14: Kriging result with value map (l) and corresponding deviation map (r). The red dots represent the observations

The variance or deviation map represents the degree of confidence in the interpolated value and therefore can be used to calculate by how much it should contribute to the result value when combined with another model of the same region but with different observations. The principle of this approach is visualized in Figure 5.15

The approach uses the inverse variances as weights [ín Martínez and Sánchez-

Meca, 2010] when fusing two grids generated from different sub-sets of observations of a region. When applied sequentially, this method successively "overwrites" the former grid, but only gradually and in regions where the new grid's variance is significantly lower. The variance maps themselves are also fused (eventually taking into account temporal decay), thus representing the confidence distribution of the new model and determining its weighting schema for the subsequent fusion step.

The process is performed for each grid cell by deriving the weight $p_{[i]}$ from its variance with

$$p_i = \frac{1}{(\sigma_i^2)^d}, \tag{5.16}$$

where σ_i^2 is the kriging variance of each grid cell, and d is an optional parameter to control the grade of weight decay relative to the variance of the model to be merged with. This factor might be adjusted according to the spatio-temporal dispersion of the given dataset. When set to 1.0, it is simply an inverse-variance weighting [ín Martínez and Sánchez-Meca, 2010].

With values and weights for each grid cell, the merged model values x_{i+1} can be derived from the current sub-model values x_i and previous model values x_{i-1} by

$$x_{i+1} = \frac{x_{[i]} \cdot p_{[i]} \ + \ x_{[i-1]} \cdot p_{[i-1]}}{p_{[i]} + p_{[i-1]}}. \tag{5.17}$$

Equation 5.17 assumes two models to be merged, which could be applied for continuous update in a real-time monitoring scenario. For the more general case with arbitrary number of models, the expression

$$\bar{x} = \frac{\sum_{i=1}^{n}(x_i p_i)}{\sum_{i=1}^{n} p_i} \tag{5.18}$$

provides the weighted result value $\bar{x}$. Respectively, its variance can be determined by

$$\sigma_{\bar{x}}^2 = \frac{1}{\sum_{i=1}^{n} p_i}. \tag{5.19}$$

In the case of real-time monitoring where the current model continuously has to be merged with new models generated from new observations, a temporal decay should be applied to the preceding model. A simple exponential decay factor f_d can be applied by

$$f_d = b^{\left(\frac{t-t_0}{r_t}\right)}, \tag{5.20}$$

with $t - t_0$ representing the time passed since the last model was generated, and b being the fraction that shall remain after time range r_t. In principle, any other covariance function (see Section 4.3) might be used to define the temporal decay rate.

Partitioning Large Models: Performance Considerations

Apart from the continuous update mechanism as assumed above, the proposed method can also be used to partition large models and apply it in a divide-and-conquer manner [Cormen et al., 2005].

Kriging comes along with a high computational complexity—caused by the inversion of the covariance matrix—of $\mathcal{O}(n^3)$ [Sun and Sun, 2015, p. 356], [Osborne et al., 2012, Barillec et al., 2011], with n being the number of samples. Considering this fact in the context of massive data load in combination with (near) real-time requirements, this can become a severe limitation of the method. Hence, when sticking to its essential advantages like the kriging variance, the merging strategy can be applied to mitigate the computational burden while delivering comparable results.

The original set of observations is separated into s subsets to which the kriging method is applied separately. The resulting sub-model grids are in the same area as the master model that contains all points. To consider all measurements in the final model, the sub-models are sequentially merged with their respective predecessor, as shown in Figure 5.16.

Alternatively, all sub-models might be calculated before they are merged in one step using Equation 5.18. This approach would, however, not provide the advantage of an immediate—albeit coarse—result. Since the linear combination of values is not equivalent to the subsequent variant, the resulting model will also differ. With the applicability for continuous updating of real-time systems in mind, only the sequential approach was investigated further here.

As is the case for any approximative solution, there is a trade-off between performance gain and resulting accuracy. As for other cases in this work, the loss of accuracy is quantified by the Root Mean Square Error (RMSE) against the master model.

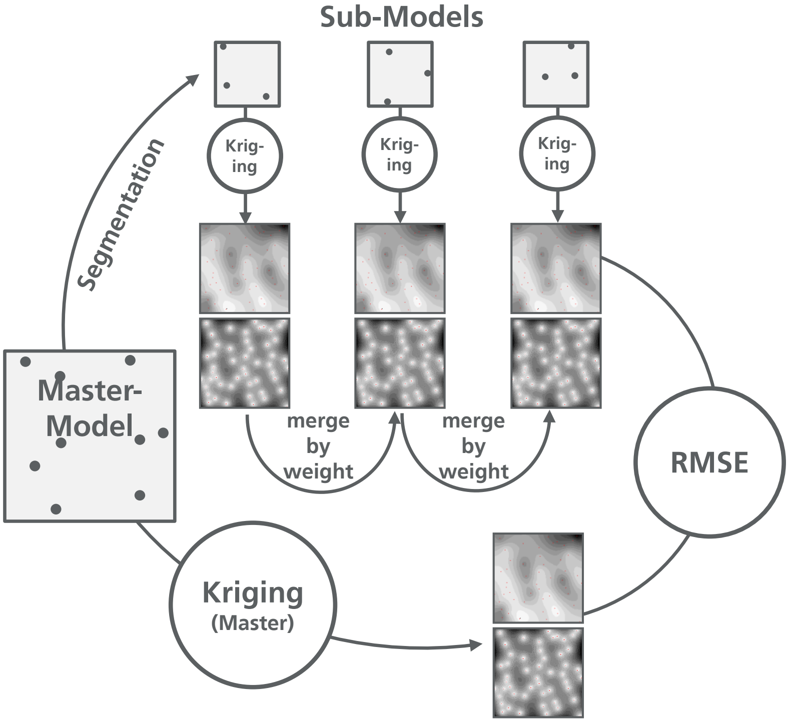


Figure 5.16: Sequential calculation schema: model partitions calculated separately and merged sequentially; the loss of accuracy induced by this approximation is indicated by the RMSE

In a spatio-temporal context, the segmentation should be performed with respect to the order of timestamps, thus representing temporal intervals per sub-model. This also applies to real-time environments where subsequent models are to be created continuously.

For a pure spatial model, the subsets of points can be generated randomly. Here, the order of sub-models does not represent the temporal dynamism of the phenomenon, but rather a utilisation level of information with associated estimated accuracy. This is also the case for the configuration as introduced below.

The segmentation and associated sequential calculation limits the potential complexity of $\mathcal{O}(n^3)$ to the size of each subset s. This can be set as a constant, but could also be dynamically adaptive to the data rate. In any case, there should be an upper bound for the size of sub-models to limit the computing complexity.

While doing so, the merge procedure itself can be costly, but grows only linearly with n and can also easily be parallelized. Thus, it is not *substantially* critical for massive data.

The theoretical computational complexity of this approach is compared to the one of the master model calculation in Figure 5.17. As can be seen from the formula given in the lower part of the figure, the reduction of complexity is achieved by removing n from cubed terms (except $n \textit{ mod } s$, which is uncritical).

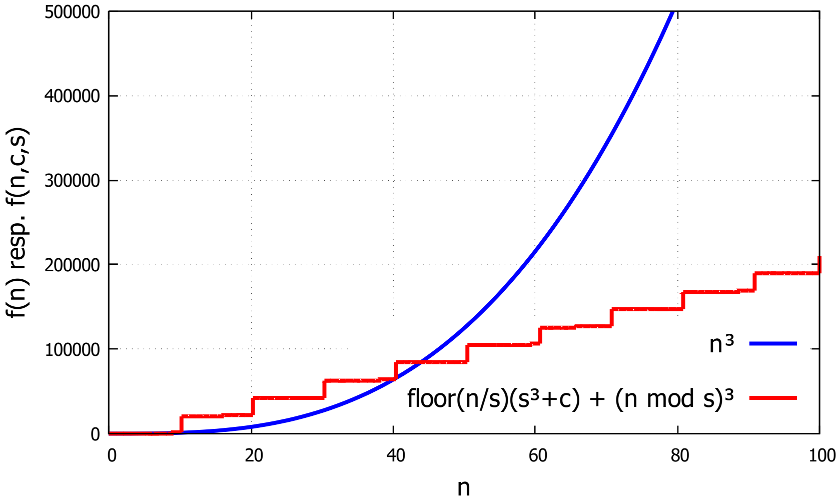


Figure 5.17: Theoretical computational complexity of master model calculation (blue line) vs. the sequential calculation method (red line); n = all samples, s = size of sub-model, c = merging effort

Assuming this merging procedure, spatially isolated or temporally outdated observations can keep their influence over multiple merging steps, depending on the decay function (Equation 5.20). This is especially helpful when no better observations are available to overwrite them. Nevertheless, with the kriging variance, the growing uncertainty of such an estimation can be expressed, which can then be considered where it appears relevant for monitoring and analysis.

Apart from some loss of accuracy, the strategy of sequencing comes along with several advantages. So it can be used to calculate large datasets with less computational effort. This can be carried out while, in principle, the advantages of kriging like the unbiased and smooth interpolation of minimum variance and the estimation of uncertainty at each position, are retained.

Given a continuous sensor data stream, this approach can integrate new measurements seamlessly into the previous model at flexible update rates. An experimental evaluation of this concept will be presented in Section 6.3.

5.4.3 Compression and Progressive Retrieval

Overview

Data compression is one key aspect of managing sensor data streams. Notwithstanding the technological progresses concerning transfer rate, processing power and memory size: they tend to be outperformed by the ever-growing amount of available observations [Gama and Gaber, 2007].

The increased mobility of sensors due to miniaturization and improved energy efficiency extends their capabilities and therefore their areas of application. On the other hand, more advanced techniques of data processing and analysis are required to exploit these new opportunities. For achieving high efficiency, compression methods should take into account the specific structure of the data they are applied to.

Sensor observations typically describe continuous or quantitative variables in multiple dimensions like latitude and longitude, time, temperature, pressure, voltage, etc. [Rodrigues et al., 2007, Blower et al., 2013]. When these data tend to be stationary in space and time, there is high potential for compression: the actual values within a confined spatio-temporal region usually cover only a small range compared to the domain represented by the respective standard data type like *floating-point number*.

In order to exploit this circumstance for compression, a partitioning of observations by spatial, temporal or other criteria (or a combination of them) into data segments is carried out. The creation of such data segments is already reasonable for storage and retrieval using spatial or spatio-temporal databases.

One central feature of the proposed concept is that it supports progressive data loading for applications that do not (immediately) need the full accuracy of the queried data. This is especially useful for environments with limited transmission rate, image resolution and processing power like for mobile computing.

For this purpose, a recursive binary subdivision of the multidimensional value space is suggested. For a given level of progression, an identical accuracy (relative to the total range of values) can be achieved for each dimension. When using a database as a sink, it is reasonable to store those data segments as BLOBs (Binary Large OBjects) indexed by the dimension(s) used for partitioning.

Queries defined by (spatio-temporal) bounding boxes then have to be processed in two steps: First, the data segments affected by the query are identified. In the second step, the data segments are progressively decoded and transmitted until the required accuracy (e.g. for scientific analysis, web mapping or mobile computing) is achieved.

Related Work

There are other compression techniques in the context of sensor observations that are discussed in literature, which are introduced in the following.

A Huffman encoding is applied to differences of consecutive measurements thus achieving high compression ratios in [Medeiros et al., 2014]. This method works very efficiently with time series of single sensors for one dimension with small changes between consecutive observations.

A more adaptive approach of Huffman encoding is introduced by [Kolo et al., 2012], where data sequences are partitioned into blocks which are compressed by individual schemes for better efficiency.

In [Sathe et al., 2013] various compression methods are introduced, mainly known from the signal processing literature. Those are restricted to one measurement variable of one sensor.

A virtual indexing to cluster measurements that are similar in value but not necessarily spatio-temporally proximate are proposed in [Dang et al., 2013]. After this rearrangement, the data are compressed using discrete cosine transformations and discrete wavelet transformations.

The compression of multidimensional signals is covered by [Duarte and Baraniuk, 2012] and [Leinonen et al., 2014]. Both works apply the Kronecker compressive sensing approach exploiting sparse approximation of signals with matrix algebra and is of high computational complexity.

Octree subdivision is applied by [Huang et al., 2008]. It exploits the proximity of values that often corresponds with spatial proximity within octree-cells. The focus here, however, is 3D visualization with specific coding techniques for colors and meshes of different detail levels instead of multidimensional continuous fields.

Requirements

The works listed above make use of the strong correlation of consecutive sensor measurements for compression. The compression method introduced here does not presume such order. Instead, it addresses the following requirements simultaneously:

- The compressed units of data are to be organized as spatio-temporally confined segments suited for systematic archiving in spatial/spatio-temporal databases
- Diverse data types, namely *Double*, *Integer*, *DateTime* and *Boolean* can be compressed losslessly
- Compression/decompression of multiple data dimensions is performed simultaneously
- Within one data segment, observations are compressed independently (no consecutive observations of single sensors tracked by their IDs are considered) and thus can handle data from mobile sensors that are arbitrarily distributed in space and time
- Data can be decoded progressively, e.g. for preview maps or applications with limited accuracy demands
- Computational cost for coding/decoding is low ($\mathcal{O}(n)$)

Principle

The principle that is applied for the compression method is derived from the Binary Space Partitioning tree (BSP tree, [Samet, 2006]). Unlike its common utilization for indexing, it is here used as compression method that is applied to each single observation in a dataset. It does not presume high correlation of consecutive observations (time series), like e.g. Huffman encoding does [Kolo et al., 2012, Medeiros et al., 2014]. Consequently, the algorithm does not need to keep track of individual sensors within a set of observations, but encodes each observation individually within the value domains given per variable dimension.

The general idea behind the design is to encode observations describing a continuous phenomenon within a (spatio-temporal) region. The focus is on the representation of the continuous field as a whole, not on the time series of individual sensors. This in mind, it appears reasonable to filter out observations that do not significantly contribute to the description of the field before long-term archiving of the data. When embedded into a monitoring system, the approach will perform best after some deliberate depletion based on spatio-temporal statistics (see Section 5.4.1 and [Lorkowski and Brinkhoff, 2015a]).

Progressive decompression can support different requirement profiles and is thus another important design feature of the approach. For some applications, it might be reasonable to give response time behaviour (at least for first coarse results) a higher priority than full accuracy after performing one step of transmission. The specific structure of the binary format supports this claim.

Binary Interval Subdivision

For each n-dimensional set of observational data, the n-dimensional minimum-bounding box over the values is determined. (In the following, the minimum and the maximum value of a dimension are denoted by *min* and *max*, respectively.) The interval $[min, max]$ will be called value domain. It is entailed in the domain that is covered by the corresponding data type.

Assuming the region of interest to be spatially and/or temporally confined and the phenomena observed to be of stationary character like temperature, there is a good chance for the value domain to be relatively small. Thus, a high resolution is achieved while requiring relatively few bits of data by using the multidimensional recursive binary region subdivision.

The principle is depicted for one dimension in Figure 5.18, where an interval is recursively partitioned by the binary sequence $0 - 1 - 1$. The circle with double arrow represents the position within the interval with its maximum possible deviation defined by that particular sequence of subdivision steps (in the following also called levels).

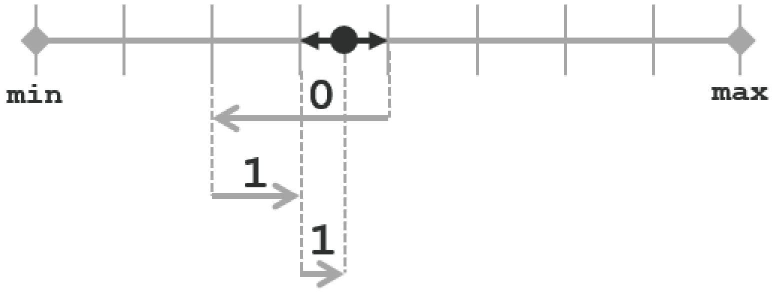


Figure 5.18: Binary space partitioning to determine a point (with maximum deviation indicated by arrows) within a value domain

As can easily be concluded from Figure 5.18, the number of necessary bits depends on both the required absolute accuracy and on the value domain.

The considerations above provide a one-dimensional perspective on the problem. For sensor data streams, this principle has to be applied to specific data types common in this context.

Supported Data Types

In a sensor web environment, the collected data can in principle be of nominal scale (e.g. type of substance), ordinal scale (e.g. Beaufort wind force), interval scale (e.g. date and time) and a ratio scale (e.g. temperature in kelvin) [McKillup and Dyar, 2010, p. 16].

In the domain of data management and programming, this kind of information is usually represented by the data types *Integer*, *Float* or *Double*, *Boolean* and *DateTime*. Within a dataset or observation epoch, the actual data range is usually only a small fraction of the range covered by the respective data type.

Since the data types mentioned above have different characteristics, they will have to be considered specifically when applying the multidimensional progressive compression.

Double/Float The compression is most straightforward for this data type. The binary tree depth n can be determined by:

$$n = log_2(\frac{d}{a}) \tag{5.21}$$

where d is the extent of the value domain $(max - min)$ and a is the accuracy or maximum deviation.

Within a multidimensional setting, the *relative* accuracy of each dimension is equal for equal n, while the absolute accuracy also depends on the size of its respective value domain.

Thus, when performing the compression synchronously for all dimensions with each step or level, as suggested here, equal relative accuracy for each dimension is achieved. This does not apply when one dimension has already reached its maximum bit depth, while others still have not (see Listing 5.1) or when particular dimensions have more than one bit per level to achieve faster convergence (see Listing 6.4 in Section 6.4).

In the case of a $Float/Double$ data type, the interval depicted in Figure 5.18 directly represents the minimum and maximum of the value domain, and the double arrow represents the accuracy or maximum deviation reached at the particular level (here: $0 - 1 - 1$).

Integer Although at first glance the data type $Integer$ ought to be *less* complex, it is in fact somewhat more difficult to handle with respect to progressive compression. First, the *fencepost error* has to be avoided when compressing/ decompressing to the last level. So if an interval shall represent *three* integer segments, as depicted in Figure 5.19, it has to be extended to *four* segments before calculating the value domain to achieve a correct representation on the scale.

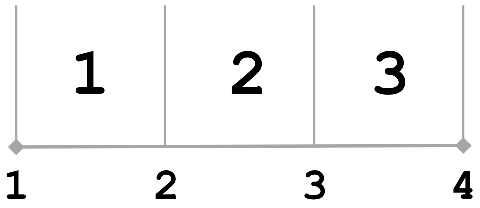


Figure 5.19: Fencepost error problem for *Integer* values

If *Integer* numbers are used for nominal scales (e.g. for IDs), coarse indications within the value domain are maybe rather useless. For that reason it might be necessary to evaluate to the complete bit depth with the first compression step. If a nominal value domain requires the maximum number of bits to be represented (see Listing 6.3 in Section 6.4), all data will have to be transmitted completely before the *Integer* value of this dimension is resolved. For more flexibility, individual bit lengths per step or level for each dimension are possible (see Section 6.4).

Boolean *Boolean* values can be seen as a special case of *Integer* with a range of 2. Consequently, only one step or level is needed to express the one bit of information (last column in Listing 6.3, Section 6.4).

DateTime Unlike the *Integer* type, the *DateTime* type appears much more complex at first glance than it is in handling. This is the case because it can be interpreted (and also is usually represented internally) as *ticks* (e.g. 100 nanoseconds) elapsed since some reference point in time (e.g. 01.01.0001, 00:00:00 h). This internal value (usually a 64-bit *Integer*) is provided by most libraries and can be used to handle the *DateTime* data type as normal *Integer* or *Double* for compression. Usually, time spans within a dataset of observations are tiny compared to the one covered by the *DateTime* type, and the necessary temporal resolution is also by far lower than that of this data type. Thus, the compression rates for this particular data type are usually high.

The data types listed above usually cover the most information found in sensor data streams. Depending on the particular structure of a dataset, differing compression algorithms might provide better efficiency.

Compression Features

Based on the common principle of compression for the different data types, the specific features facilitated by that principle will be set out in the following.

Parallel compression of all Dimensions One central feature of the proposed compression format is the progressive retrieval of sets of observations with increasing accuracy with each step or level. The general format is shown in Listing 5.1 which displays the compression format for seven dimensions of one observation.

```
i     o
dxyztvn
-------
1010111
101000
100001
010001
111011
0010 1
00 0 1
01 0 0
10 0 1
0  1
1
0
```

Listing 5.1: Binary compression format for progressive sensor data storage (column names and values to be read vertically downwards); after its name, each column contains the binary representation of the value dimension with increasing accuracy per step

Each column entails one value dimension and each row represents one level of progressive coding/decoding. The bitstream of a particular dimension terminates at the level where its preset resolution/accuracy is reached. For the data type *Boolean* (right column: *on*) this is already the case after the first step or row.

Unlike the structure displayed in Listing 5.1 for visualization, the actual binary format does not contain blank positions, but only the data bits. Therefore, for decompression it is necessary to consider the format structure to have each bit assigned to the correct dimension.

Due to its general structure, with increasing row numbers this format tends to decrease in data volume per row and finally contributes to the accuracy of the dimensions with highest predefined resolutions only.

Flexible Bit Length per Row Given the structure described above can lead to a situation where a particular dimension might not be determined at desired accuracy until the last row is reached. Most of the data might have been transmitted unnecessarily because a low accuracy would have sufficed for the other dimensions. This situation might particularly be the case for IDs (first column in Listing 5.1 and 5.2) or nominal scales. It might be indispensable to receive their exact value at an early stage of the stepwise transmission.

As solution for this problem, the bit lengths per row can be set individually for each dimension (see column *id* in Listing 5.2). Thus, the value of a dimension can converge much quicker towards its actual value with each step. In the extreme case, the exact value can already be provided with the first step of transmission (as it is *always* the case for binary values). This option can be useful when the IDs of observations are needed immediately for visualization or mapping with other data sources.

```
i       o
d  xyztvn
---------
111010111
10001000
10100001
00010001
   11011
   010 1
   0 0 1
   1 0 0
   0 0 1
   1
```

Listing 5.2: Binary format with flexible bit length per dimension; here, dimension *id* is coded with three bits per row reducing the necessary rows to four instead of twelve

Progressive Decompression As a consequence of the special data structure introduced here, the decompression process must permanently keep track of the actual bit configuration and the number of bits processed so far. With each new row transmitted, there is an improvement of accuracy (the factor depending on the number of bits per row) for each dimension. In an environment with bandwidth restrictions, this progressive method provides immediate coarse results, e.g. for visualization. With the last step, the data is transmitted completely lossless according to the predefined resolution. This is not always necessarily the best choice since the data might not be needed immediately in full accuracy but rather within shorter transmission time (responsiveness). The transmission can therefore be aborted at any level.

An experimental evaluation of the compression concept set out here is carried out with buoy data in Section 6.4.

5.5 Generic Toolset for Variation and Evaluation of System Configurations

So far in this work, the purely operational aspects of the monitoring like sampling, interpolation, sequencing and compression have been paid attention to. Yet, beyond this perspective, also the quality and efficiency of different monitoring scenarios are subject to this thesis.

Variations of methods and associated parameters will affect the output of a simulation scenario. A continuous optimization of the whole process can only be carried out with appropriate output performance indicators expressing both quality and efficiency.

In this section, a general concept for systematic variations of methods and parameters and their effects on output parameters is introduced. It abstracts from the particular algorithm at hand and provides a generic toolset for simulation environments.

5.5.1 Context and Abstraction

As already argued in Chapter 3, a monitoring system should be designed to provide sufficient model results with the least resources possible. The aspect of resource requirements and performance indicators of a monitoring is illustrated in Figure 5.20.

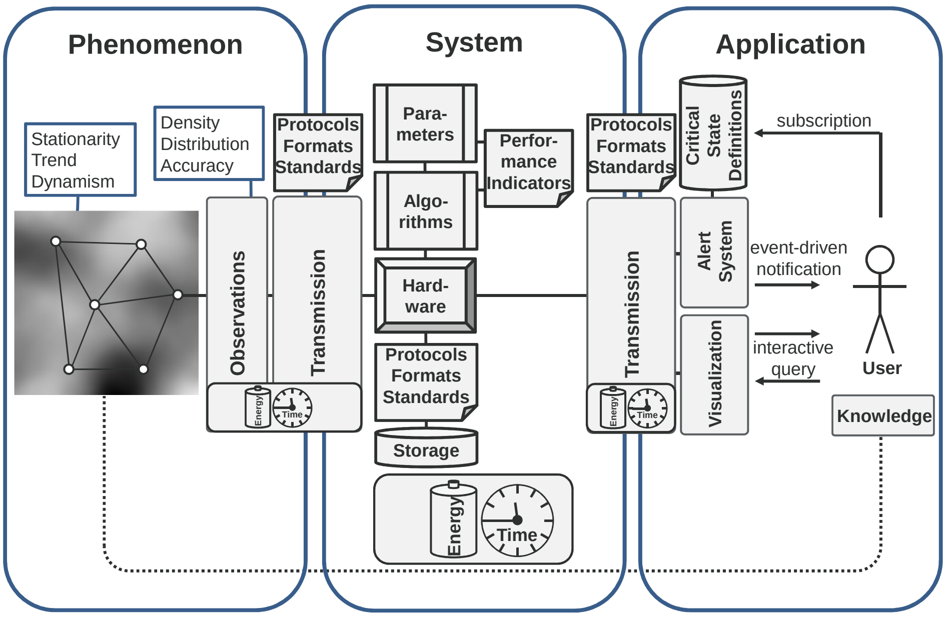


Figure 5.20: Elements of monitoring considering the limited ressources time and energy

The phenomenon needs to be observed and the observational data have to be transmitted to the system. The system processes and archives the data in a way that provides information of higher generality, abstraction and therefore of higher value to applications. Specifically, the improvement takes place on several levels:

- coverage
- accuracy
- density
- interoperability
- interpretability
- usability

In other words: by deploying resources for computation and transmission (time and energy), a monitoring system transforms raw observational data to valuable information according to the aspects listed above. Using these resources efficiently is the obligation of any monitoring system.

Applications of various kinds can make use of such higher-level services provided by the system. Details about interpolation can thus be decoupled from the application logic [Taylor et al., 2009, Evans, 2003]. The concept of a field data type [Liang et al., 2016, Camara et al., 2014] is one of the key features to achieve this goal.

The overall objective is to provide knowledge about the phenomenon that is in some way useful. Since resources for such a monitoring are limited (see Section 3.3), the challenge is to find some good compromise between cost and benefit.

The means to establish such a monitoring are sensors, communication networks, computers, algorithms and their associated parameters, and standards for transmission and interoperability, as depicted in Figure 5.20. The hardware-equipment of a monitoring system should be configured following the principles formulated in Section 3.4 and balancing the factors featured in Figure 3.2.

The effectiveness and efficiency of such a monitoring system need to be estimated in the planning phase, but also need to be evaluated and improved when the system is operating. The most crucial decision is about accuracy, density and distribution of observations (see Figure 5.20; also Section 3.3.2 and 5.3.2). At this stage, the degree of knowledge about a phenomenon is determined since even the most sophisticated processing methods cannot compensate insufficient sampling.

In order to be processed for a whole region, the observational data need to be transmitted and collected within a sensor network. Appropriate transmission protocols and data formats should be used in order to minimize time and energy expenses.

Once the data are available in the central system, complex operations like spatio-temporal interpolation can be performed. Hardware, algorithms and the amount of data determine the expense in time and energy here. Variation of algorithms and adjustment of parameters can improve quality and efficiency, which will be stated by performance indicators. Persistent storage preferably is carried out on a database, supporting spatio-temporally referenced data and fostering efficient retrieval; compression (Section 5.4.3) reduces storage space and transmission effort.

The main processing component of the monitoring system (named 'System' in Figure 5.20) is adding value to the observations in the sense that gaps are filled and the provided format is by far more interoperable than the original sensor data. In the ideal case, the phenomenon is presented as continuous model with a good estimation of the variable at arbitrary positions in space and time within the observed area.

Kriging also provides the estimation variance for each position, which might be used for smooth updates and performance improvement (see Section 5.4.2), but also as adaptive filter (see Figure 5.13). Being available in this form makes it by far easier to navigate the data interactively or access it from applications via a web service.

More complex services like alert system based on aggregated data (e.g. notifying about an exceeded daily threshold for a region) can be constructed when such an infrastructure is available. The desired quality of such services determines the minimum costs and efforts necessary to establish them and keep them operational. An experimental setup as introduced in this work can significantly contribute to improve efficiency and reduce these costs.

Following the objective of balancing cost and benefit of the system, an iterative optimization is carried out to get the best possible results from limited resources. In the context of computing systems, the expenses in time and energy are most relevant to be considered to achieve a result of particular quality.

Time for processing is critical when hardware power is limited, whether for financial or technical reasons. Energy is most critical for small battery-powered systems as well as for big systems like mainframe or cluster computing systems. Efficient processing can significantly reduce costs in both cases.

This section focuses on potentials for optimization by variation of algorithms and parameter settings. To *systematically* and *reproducibly* evaluate the efficiency of each variant, a generic concept for a quantification of the following aspects is needed:

- workload
- resources
- output indicators

The workload is the computational effort that is necessary to process a particular input dataset by a particular algorithm with a particular corresponding parameter set. The resources represent the computational hardware that is available for this task. The indicators quantify the output quality and other benchmarks like expenses in time and energy that are necessary for a particular constellation of the two other components. Algorithmic optimization and parameter tuning affects the quantity of workload and therefore also the output indicators [Beven, 2009, p. 11]:

> For each combination of parameter values, we can calculate a model response.

Doing so while overlooking the effects that different parameter settings have on the different output indicators can then be regarded as an evolutionary process towards better and better solutions [Gandibleux et al., 2004].

One important intention behind the proposed model is to quantify efficiency improvements independently from the hardware configuration that the simulation is currently calculated on. When this quantity is combined with a concrete hardware configuration, expenses in time and energy can be derived.

Especially for wireless systems, the estimation of the actual temporal and energetic expenses on a particular hardware constellation can be crucial in the planning phase. Such a transfer of processing expenses can only be carried out with a generic concept for computational workload, which is introduced in the next section.

The automatic variation of algorithms and corresponding parameter settings is the second objective necessary for systematic evolutionary improvement. In order to handle and evaluate configurational settings of arbitrary complexity, a generic hierarchical structure is introduced in Section 5.5.3.

Together, the two components form a powerful toolset to systematically test and evaluate numerous configurations concerning hardware and algorithms in complex processing scenarios.

5.5.2 Computational Workload

When processing tasks are so complex that their execution might exceed critical resource thresholds, the resource requirement for a particular workload (or *job*, meaning “information-processing task” [Ferrari, 1978, p. 225]) is often specified by execution time on a particular machine. It is easy to obtain and for many purposes provides a sufficient estimation.

This kind of metric has, however, several drawbacks because it strongly depends on the system it was actually measured on. Given the hardware specifications of this system, it might appear easy to predict the execution time for a system with different hardware. In practice, however, it can not simply be concluded that, e.g., double CPU clock speed means half execution time etc. Many other factors like bus frequency, amount of memory, number of processor cores, and implementation details of the program will also affect the overall performance [Fortier and Michel, 2003].

From a practical viewpoint, it might not pay off to consider all of those factors right away. Instead, a model should initially contain only the most influential factors and be equipped by additional factors only if it proves to be inadequate [Lavenberg, 1983, p. 8]. With respect to this principle, the properties to be considered in this work are the CPU speed, the number of logical processors and the capability of critical code sections to run on multiple threads.

In order to predict the performance on different platforms, the central objective is to describe a particular computational workload in a way that does not strongly depend on the execution environment. The central idea to achieve this is to decouple the logical instructions from the physical resources like CPU speed [Ferrari, 1978, p. 225]. This separation makes it possible to estimate the processing time for a properly described workload without actually having to execute it on the particular machines.

This can be an indispensable information when some particular response time has to be granted for a monitoring service and sufficient hardware must be deployed. Such considerations can even be more important for wireless sensor networks since workload quantity is, at least to a certain degree, proportional to energy consumption on identical hardware.

As a consequence of the considerations above, the workload model and the execution environment or hardware system model have to be defined separately, but with associations to each other according to logical and physical properties and resources. A prototypic realization of this general concept is given by Figure 5.21.

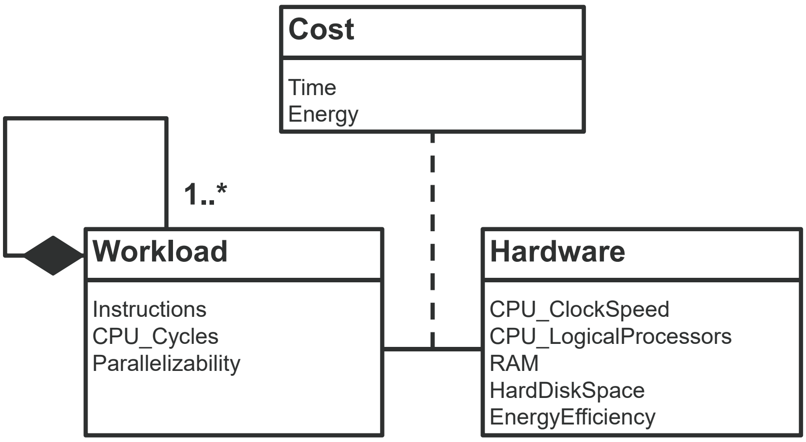


Figure 5.21: Generic structure to quantify computational cost: class *Workload* represents the machine-independent workload quantity units (as composite aggregation [Larman, 2001, p. 414 ff.]) with differentiation between parallelizable and non-parallelizable sequences; class *Hardware* specifies performance-relevant properties of a processing unit; class *Cost* determines the expense in time and energy resulting from such a given association between *workload* and *hardware*.

The central feature of this structure is the systematic differentiation and aggregation of code segments according to their parallelization capability. This is of primary relevance because modern computer systems increasingly utilize parallelization [Cormen et al., 2005], and, consequently, so do complex applications like spatial interpolation [Pesquer et al., 2011, Wei et al., 2015, Jardak et al., 2010]. In principle, the same pattern is applicable in scenarios that work with graphics processing units (GPU) [Henneböhl et al., 2011].

In order to obtain an abstract and machine-independent description of a particular workload, all of its logical instructions need to be counted while keeping track of their capability for parallelization. Within a complex computing task, there will usually be sequences that are implemented using parallelization, but also ones which do not, for example, because it is not possible (serial algorithms) or because the expected efficiency gain does not justify the implementation overhead.

By defining the total computational cost as a composition of sub-portions, as indicated by the UML class diagram (Figure 5.21), it is possible to divide the entire workload into portions that can be specified individually according to their parallelization capability.

Assuming the temporal effort for a particular hardware configuration as metric, these portions sum up to the entire workload by

$$t = \sum_{i=1}^{n} \frac{gc_i}{f \cdot fc \cdot thr_i \cdot thro_i}, \tag{5.22}$$

where gc_i is the computational workload of the portion i of the algorithm, expressed as unit *gigacycles* (billion processor cycles), thr_i is the number of threads this portion can be calculated with (will be *one* for non-parallelizable parts) and f is the CPU clock frequency in GHz. In addition, to take into account product-specific differences in the number of instructions that can be processed per cycle, the factor fc is introduced. It might either be determined experimentally or derived from product specifications. With $thro_i$, the overhead of multithreading is also considered for each portion of code. It is set to 1.0 if multithreading is not carried out.

Equation 5.22 represents the class *Cost* from Figure 5.21 by combining the machine-independent parameter gc_i with the other, machine-dependent parameters.

The quantity *gigacycles* might be obtained or estimated in different ways, e.g., by using external performance evaluation tools. Integrating this task into the development process—i.e. into source code—provides maximum control and extensibility [Smith, 2007, p. 419 f.]. For that reason, the approach was also chosen for the framework introduced here.

For the experimental evaluation as set out in Section 6.5, the quantity *gigacycles* is obtained by the C++ function *QueryProcessCycleTime* that is imported as external code to the C# environment. Although the term *time* within the function name indicates the physical unit, it actually provides the number of all CPU cycles of the calling process since it started. The function sums up the cycles from *all* running threads, so this is the value that is to be stored as attribute of the *Workload* item as defined in Figure 5.21, with the *Parallelizability* attribute set to *true* if implemented accordingly.

Given a set of *Workload* objects that were deliberately registered with respect to the capability of parallelization of the respective code, it is straightforward to translate this structured quantity into processing time on a particular hardware (Equation 5.22).

While consumption of *time* is crucial for complex processing tasks and real-time monitoring applications, the consumption of *energy* is especially critical for battery-operated devices as used in wireless sensor networks.

To check the operability of such a system and to optimise it according to energy efficiency, it might be necessary to estimate the energy consumption for a particular hardware configuration. Therefore, a rough estimation of the total energy consumption w, e.g. stated in the unit *nanojoule* that is necessary for a particular process, can be given by the similar equation

$$w = \sum_{i=1}^{n} gc_i \cdot w_i, \tag{5.23}$$

where w_i is the amount of energy consumed per gigacycle in each portion gc_i of the algorithm.

Individual values for each process portion can be considered where different amounts of energy per cycle do occur, eventually depending on whether it is parallelised or not. As already mentioned, there might also be portions of an algorithm that can be delegated to a graphics processing unit (GPU) or field programmable gate array (FPGA) [Liu et al., 2012], which would eventually call for individual specification.

The aspect of energy consumption is not covered beyond this conceptual level here. However, in the context of wireless sensor web scenarios it appears reasonable to also simulate energy consumption per processing unit in order to find efficient monitoring strategies. Given the general structure as described above and as formalized in Figure 5.21, the model can easily be extended with respect to energy consumption.

The quantities given by Equations 5.22 and 5.23 can only be seen as approximations since there are many aspects which can blur such calculations. A closer consideration of following factors might therefore be necessary when the concept is to be refined (see also [Fortier and Michel, 2003]):

- hardware design: CPU, memory access, pipelining
- multithreading management overhead (synchronisation, etc.)
- processing portions delegated to a GPU
- programming language (e.g. garbage collection)
- compiler optimizations (e.g. JIT compiler effects [Nagel et al., 2005])
- operating system
- different number of instructions per clock cycle

Where necessary, these blurring influence factors can be estimated and included into the equations. In summary, the concept of a machine-independent workload metric is at least a rough but systematic approximation necessary to test algorithmic variants with regard to several performance indicators for different system configurations. It is an important contribution towards iterative optimisation, especially for real-time monitoring or distributed systems like wireless sensor networks.

5.5.3 Systemantic Variation of Methods, Parameters and Configurations

For a monitoring scenario as set out in this work, there is a variety of method variants, parameters and configurations that need to be evaluated with respect to their performance. One possible approach for testing different configurations is to vary *one* parameter while leaving other parameters fixed and regard the resulting series according to some evaluation metric and thus determine the best variant of this particular parameter [Sun and Sun, 2015]. Repeating this procedure for n parameters reveals a set of parameter configurations which might be considered appropriate for the given process. The number of variants to be tested is therefore the *sum* of variants per parameter.

But there is a fundamental problem with this approach: There has to be an initial configuration for *all* parameters for which the variation of *one* parameter per testing epoch is performed. This initial configuration has often to be chosen arbitrarily. Favourable constellations of parameters might therefore remain undetected because they are not tested in this scenario.

Alternatively, *all* possible constellations of parameter settings can be considered. The total number of tests to be executed is then the *product* of the number of variants per parameter instead of their *sum*. This might of course place a considerable burden on testing scenarios.

For example, varying only ten parameters by only ten values or options each will result in 100 configurations to check for [Jorgensen, 1994, p. 61]. However, it systematizes the process and makes it by far less arbitrary. The results generated by such a systematic survey of variants allow for more systematic and extensive analyses and therefore promote a deeper understanding of the whole process that is evaluated.

In complex systems like spatio-temporal analysis tools, monitoring environments or simulation frameworks, algorithmic variants and associated variable parameters tend to increase over time. As a consequence, manual or semi-automated setting and evaluation of variants, e.g. by configuration files, become increasingly cumbersome, error prone and arbitrary. A generic software solution for this problem is sketched as UML class diagram in Figure 5.22.

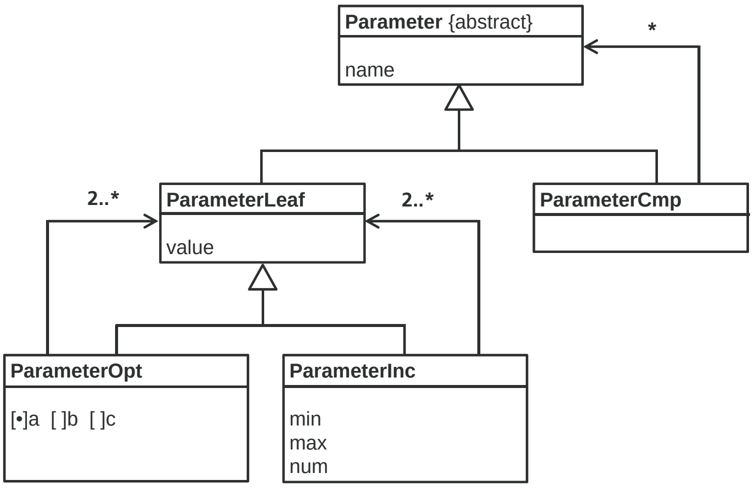


Figure 5.22: UML class diagram for generic organisation of configuration variants based on the composite pattern [Larman, 2001]: class *Parameter* as abstract concept and *ParameterCmp* as container for structured organization; class *ParameterLeaf* representing the instance actually containing the value, concretized as option (class *ParamterOpt*) or as increments within an interval (class *ParameterInc*)

The composite pattern supports hierarchical organisation and polymorphic treatment of whole-part-relationships of objects [Larman, 2001, Szyperski, 2002]. We exploit this capability of the pattern here to define a generic structure which can handle arbitrary complex configurations of parameter variants in a uniform way. The class *Parameter* represents an abstract concept that can be instantiated as a list of mutually exclusive options (class *ParameterOpt*), an interval division comprising integer or float type increments (*ParameterInc*), the corresponding enumeration or numerical value itself (*ParameterLeaf*) or a named composite of multiple parameters (*ParameterCmp*).

This pattern is chosen for its capabilities to reflect the complex hierarchical parameter structure which is present in many systems. By using recursive polymorphic function calls, all possible configuration variants can be created and iterated [Szyperski, 2002, p. 83 ff.], [Mellor and Balcer, 2002, p. 227 ff.], [Mellor and Balcer, 2002, p. 255 ff.].

With this structure it is also possible to organize parameters in sub-trees that describe logical units within the modelled environment. So a hardware configuration as described in Section 5.5.2 can be subsumed as composite type *ParameterCmp* containing parameters for number of CPU cores, CPU clock speed and RAM. Variation can then be carried out by each parameter (e.g. 1, 2, 4, 8 CPU cores; 1, 2, 4 GiB RAM, 1.2, 2.0, 2.6 GHz CPU clock speed) or, alternatively, by predefined named configuration sets (e.g. Raspberry Pi® 3 Model B, Dell Precision® Tower 5810) and the corresponding detail information parameter set. The variation can then take place by switching between those named subsets.

With this structure, variations of preset hardware configuration can be carried out just as easily as any other set of parameters within a simulation. An estimation of processing time for machine-independently defined workloads can then be carried out for each of those configurations. This functionality can be important when considering the hardware equipment for a planned monitoring system.

There might also be constellations where environmental conditions that affect data transmission—like meteorological parameters—should take part in the systematic variation. Analogous to hardware configurations, such constellations can also be modelled and handled as named composite subsets.

As already mentioned, the guiding idea of the structure described above is to automatically generate the list of all possible configurational variants instead of having to update such a list manually with every added or removed parameter option. In order to generate series of simulation variants based on this structure, it has to be integrated into the superordinate flow control of the simulation framework.

For systematic evaluation of those variants, their output performance indicators are stored in tabular form using the naming conventions of the identifiers for algorithms and parameters (see Tables 6.2 and 6.4 in Section 6). The same conventions are used for the directory structure the output data is stored to. The indicator log file relates configurational variants to performance indicators for each process run and thus facilitates deeper and more systematic analysis.

5.5.4 Overall Evaluation Concept

The main objective behind the concept proposed so far is to organize variants of analyses or simulations and evaluate the different outcomes by one or more performance indicators. It depends on the particular application which input properties and which performance indicators are to be taken into account for such experimental series.

Table 5.1 provides a representation of the general concept of such a relation. For a wider perspective, it is extended by properties and indicators that can be considered reasonable in the context of environmental monitoring.

	Output Indicators / **Input Properties**	Time	Energy	Data Volume (MB)	Transmission Effort	Accuracy	Compression (Rate, Loss)
Env.	Phenomenon Dynamics					●	
	Sampl. Rate/Distr.		●	●	●	●	
	Transmission Medium	●	●		●		
Hardware	Sensor Energy Efficiency		●				
	Comm. Bandwidth	●			●		
	Comm. Energy Efficiency		●		●		
	CPU clock speed	●	●				
	CPU logical processors	●	●				
	RAM	●	●				
	Storage	●	●				
	Computational Efficiency		●				
Data	Input Data (Amount/Format)	●	●	●	●		
	Data Density (Raster/Vector)	●	●	●	●	●	
	Compressibility	●	●	●	●		●
Algorithm	Method Set	●	●	●	●	●	●
	Parameter Set	●	●	●	●	●	●
	Parallelization Capability	●	●				
	Compression Method	●	●	●	●	●	●
	Indexing Method	●	●	●			

Table 5.1: Input properties (arranged by categories environment, hardware, data and algorithm) and output indicators of complex computing systems; their interdependencies are indicated by dots

Input Properties contains all the items and properties that constitute the environment and monitoring system as a whole: the environmental conditions that influence the monitoring, the hardware, the methods, tools, datasets and formats that are used to generate the model. For a given scenario of input properties, the *Output Indicators* represent the metrics that can be used to evaluate the whole process chain. The interdependencies between those items are specified in the following.

Environment The observed phenomenon can be described by its dynamism in space and time. Leaving all other elements of the monitoring system unchanged, it only affects the accuracy of the model. If the phenomenon has more dynamism than is covered by the sampling layout, this will affect the model accuracy.

Changing the sampling rate affects the effort that is necessary for sensing, computation, transmission and storage, but it also changes the output accuracy. The transmission medium entails atmosphere, topography and also potential sources of interference which affect the transmission effort that can also be associated with energy effort. If the transmission signal strength is deceasing [Ahmed et al., 2012] and has to be repeated due to errors indicated by the protocol, also temporal delay may occur.

Hardware The properties of the hardware involved in monitoring are listed in the next group of Table 5.1. Leaving factors like workload or algorithm unchanged, the energy demand is affected by the efficiency of sensors, communication devices and processing units. The throughput per time unit depends on hardware specifications and communication bandwidth while the effort for transmission according to time and energy depends on the efficiency and bandwidth of the communication devices.

When plenty of harddisk storage is available, time and energy performance can be increased by providing multiple indexes, controlled redundancy and preprocessed data (e.g. discretization of continuous fields by raster grids , see Section 3.2.3 and Chapter 7).

Data From the data perspective, it is the amount and format of incoming sensor data that affects most of the output indicators. The data *density* in the context of the *Data*-group does not refer to the raw observational data, but rather addresses archiving and retrieval. Vector data can be thinned out deliberately while minimizing information loss under the presumed interpolation method (see Figure 5.13). Raster grids can be provided statically or dynamically in different resolutions depending on the requirements.

Having efficient transmission and storage of data in mind, the compressibility is another important factor. The structure of observational data of continuous phenomena allows for good compression rates and progressive retrieval (see Section 5.4.3). This reduces the data volume, but goes along with extra effort for compression and decompression, which might affect response time. This also might affect energy resources, since compression/decompression is less expensive than data transmission [Appice et al., 2014]. The compression rate depends on both the method and the data. Sophisticated lossy algorithms for raster grids achieve high compression rates with small loss of accuracy [Press et al., 2007].

Algorithm In many cases, the choice and configuration of *Algorithms* is the most obvious way to influence the output indicators. The set of methods with their associated parameters can have impact on any indicator. It is also significant how the particular algorithm was implemented. The parallelization capability is depending on both the implementation and on the hardware. It primarily affects processing time and therefore response time. Because it is entangled with data volume and accuracy, the compression method can affect all indicators. Indexing reduces search operations and therefore saves time and energy while increasing necessary data volume.

As has been shown above, the interdependencies between input properties and output indicators within a monitoring system are manifold. For real world scenarios, they might be more complex than is indicated by Table 5.1. A systematic inventory like this, however, does support the process of decision-making when establishing or auditing an environmental monitoring system. Beyond the accuracy-centred model evaluation as suggested in [Beven, 2009, p. 3], it also takes environmental, technical and resource-based matters into consideration. Thus, it can help to systematically evolve a monitoring system towards better quality and efficiency.

5.6 Summary

In this chapter, various methods and algorithms have been set out that are necessary for an efficient monitoring of continuous environmental phenomena.

Beside the central task of interpolation, also the generation of continuous random fields, the sampling of such fields and the merging of sub-models (for efficiency and smooth differential updates) have been addressed.

The proposed features constitute a powerful simulation environment that is capable of automatically testing manifold modes and configurations of environmental monitoring. In order to facilitate this task of carrying out and evaluating experiments with multiple configurations, a generic software solution has been proposed. It automates configurational permutation and relates parameter settings to output datasets and output indicators in order to systematize the evaluation process.

The simulation experiments that are introduced in the next chapter have been carried out to evaluate some of the crucial methods that were introduced here. The tool for systematic variation and evaluation has been used extensively in these experiments.

Chapter 6

Experimental Evaluation

6.1 Minimum Sampling Density Estimator

This section provides an outline of the experiments that were carried out in order to evaluate the formula for the minimum sampling density as it was deduced in Section 5.3.2. A simulated random sampling is therefore carried out on synthetic fields. By varying the number of samples around the deduced optimum, its validity is inspected by comparing the RMSE of each interpolated model.

6.1.1 Experimental Setup

To check the validity of the approximation of the necessary minimum sampling density, the method of kriging is applied to sets of observations, varying in number, performed on different synthetic continuous fields. Within one set, the observations are randomly and uniformly dispersed over the n-dimensional region of interest.

The differences (RMSE) between the reference field and the one derived from the interpolation are compared. The experiment will be carried out on different kinds of random fields, which will be specified before the respective results are presented.

6.1.2 Results

As first reference, a two-dimensional field is generated by

$$f(x,y) = sin(x) \cdot sin(y). \tag{6.1}$$

The resulting raster grid of $150x150$ pixels in greyscale levels is depicted in Figure 6.1.

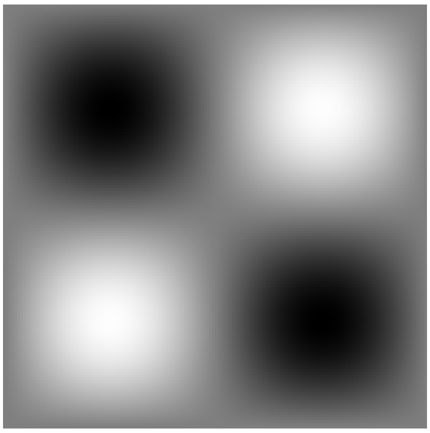


Figure 6.1: Two-dimensional sine signal as raster grid

With Equations 5.3, 5.8 and the extent of 1λ in each spatial dimension we get

$$2\frac{1}{\frac{1}{4}} \cdot 2\frac{1}{\frac{1}{4}} = 64 \tag{6.2}$$

as approximate minimum number of samples necessary to capture the pattern for Kriging. We apply seven sampling sets from 25 up to 115 observations, increase by 15 observations with each step, normalize it to the calculated value of 64 and plot this quotient against the RMSE between reference model and derived model. For convenience, this value is normalized to the highest one in the series. The parameter *range* is also added to the diagrams discussed in the following. It is derived from the variogram fitting procedure (see Section 5.3.5) and is normalized to the theoretical value determined by Equation 5.3.

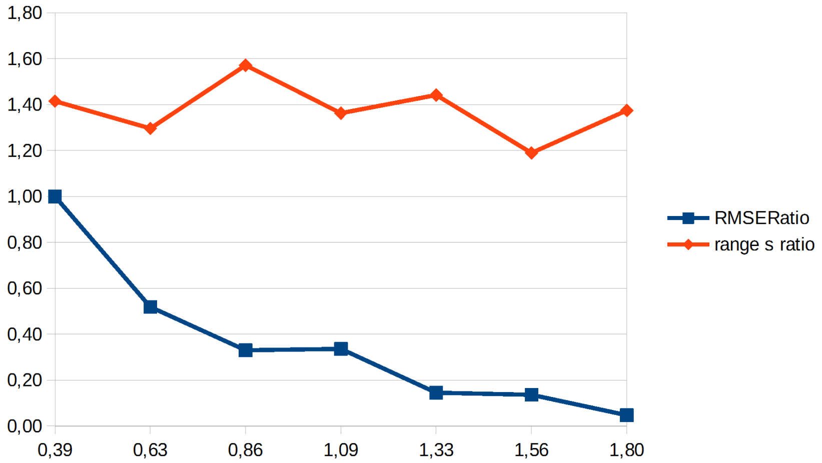


Figure 6.2: Sampling variations applied to a two-dimensional sine signal with the ratio of sampling normed to the derived value on the abscissa, and the ratios of RMSE and *range* normed to the initial value (RMSE) and to the value of the generated field ($range\_s$)

As can be seen from the RMSE graph of Figure 6.2, a noticeable degree of saturation is achieved when the quotient approaches the value of one, which represents the minimum number of samples of 64 as computed by Equation 5.8.

Extending the sine signal by a third dimension reveals a similar pattern, as can be seen in Figure 6.3. In this case, the number of samples each epoch is normalized to is

$$2\frac{1}{\frac{1}{4}} \cdot 2\frac{1}{\frac{1}{4}} \cdot 2\frac{1}{\frac{1}{4}} = 512. \tag{6.3}$$

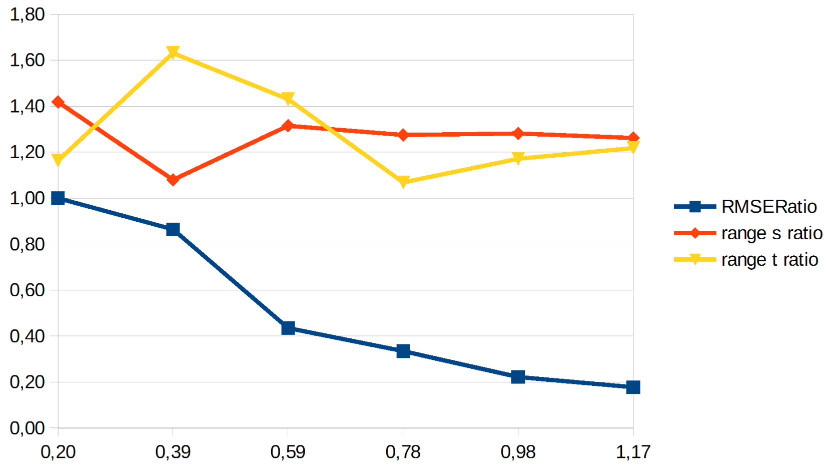


Figure 6.3: Sampling variations applied to a three-dimensional sine signal with the ratio of sampling normed to the derived value on the abscissa, and the ratios of RMSE and *range* normed to the initial value (RMSE) and to the values of the generator $(range\_s, range\_t)$

Having used the separable variogram model for interpolation, the parameter *range* is separately estimated for the temporal dimension. Other models might also be applied here (see Equations 4.8, 4.9, 4.10, p. 56), but this is out of the scope of this evaluation. For the spatial dimension we assume this parameter to be equal for each direction; otherwise anisotropy would have to be introduced [Webster and Oliver, 2007].

The sampling on sine signals primary was carried out for the reason of transfer of concept of the Nyquist-Shannon theorem from signal processing to geostatistics (see Section 3.3.2). After the validity for periodic signals was shown, it was applied to continuous random fields as depicted in Figure 6.4.

Figure 6.4: Two-dimensional synthetic random field generated by a Gaussian covariance function

Given an extent of 150 and a range of 30, generated by a Gaussian covariance function (see Section 5.3.1), the number of necessary observations is calculated by

$$2\frac{150}{30} \cdot 2\frac{150}{30} = 100. \tag{6.4}$$

In the diagram (Figure 6.5), the effect of a saturated error quotient can be found near the abscissa value of 1.0 that corresponds with the estimated minimum sample size.

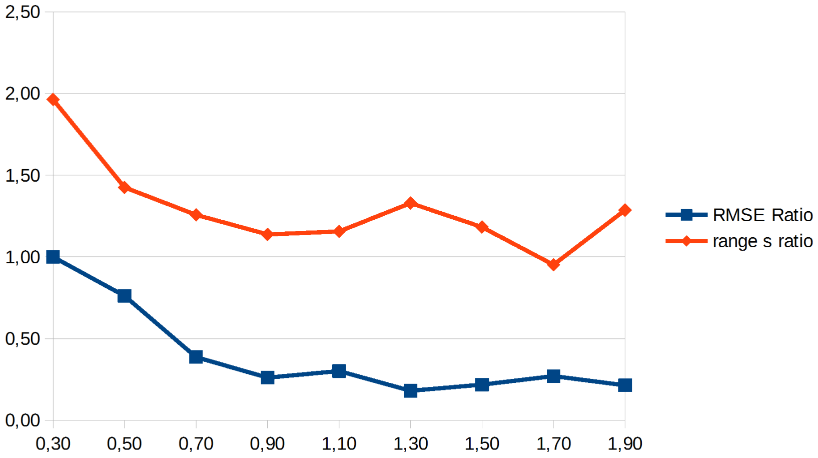


Figure 6.5: Sampling variations applied to a two-dimensional random field with the ratio of sampling normed to the derived value on the abscissa, and the ratios of RMSE and *range* normed to the initial value (RMSE) and to the value of the generator ($range\_s$)

In this case, the similarity between the RMSE curve and the range s ratio curve is striking, indicating that the accuracy of the estimation of the parameter $range$ corresponds with the accuracy of the whole derived model.

This effect is less obvious in Figure 6.6, which represents sampling epochs performed on a three-dimensional random field. There is also a generally higher ratio between estimated and actual range parameter here indicating an increased uncertainty of estimation due to the higher complexity of the phenomenon. The saturation effect of the RMSE when the sample size approaches the number estimated by Equation 5.8 can nevertheless also be identified quite clearly.

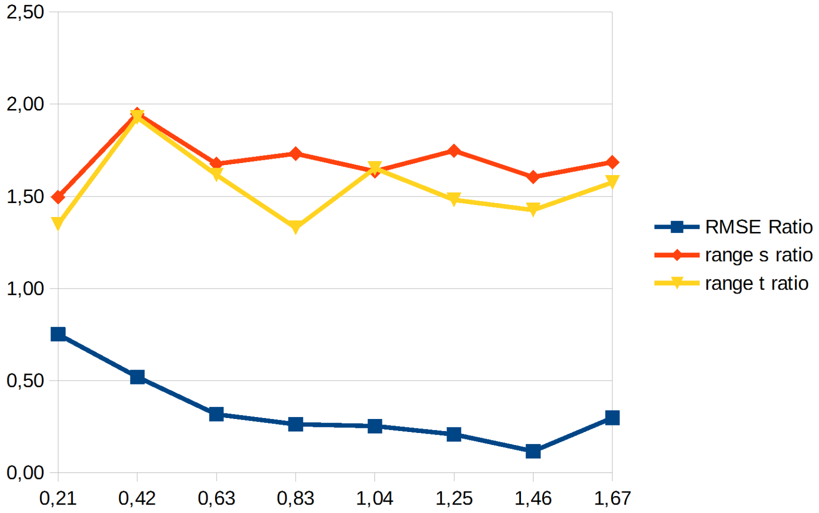


Figure 6.6: Sampling variations applied to a three-dimensional random field with the ratio of sampling normed to the derived value on the abscissa, and the ratios of RMSE and $range$ normed to the initial value (RMSE) and to the values of the generator $(range\_s, range\_t)$

6.1.3 Conclusions

The experiments have corroborated the overall validity of the formula for minimum sampling density. It can thus be used to estimate the observational effort for any setting where the central geostatistical parameter $range$ is known for all involved dimensions. It assumes a uniform random distribution of sample positions and can therefore only provide an approximate estimation. For the simulated monitoring scenarios it is of great value since it relieves the process of sampling of arbitrariness and makes experiments on models of differing dynamics (and therefore differing values of *range*) comparable. We will make use of this formula in the subsequent experiments to reduce effects resulting from insufficient sampling or oversampling.

6.2 Variogram Fitting

Geostatistical interpolation is carried out by applying a particular covariance function with its associated parameters to generate a covariance vector for each position to be estimated. This vector expresses the correlation to each single observation by using its (n-dimensional) distance to the interpolation point as input variable for the covariance function. Based on this structure, a linear regression is performed to find optimal weights for each observation [Armstrong, 1998, Webster and Oliver, 2007, Oliver and Webster, 2015].

In order to adapt to a particular set of observations, the parameters of the variogram model are adjusted to fit the corresponding experimental variogram (variogram fitting, Section 5.3.5). The appropriateness of the variogram model and its associated parameters determines the quality of the interpolation and is therefore decisive for the whole process.

In an experimental setting with synthetic continuous random fields as given here, the quality of interpolation and therefore the quality of variogram fitting can be expressed as RMSE between the reference model and the one derived from interpolated observations. This is the setting that is applied in this section to identify suitable configurations for variogram fitting.

6.2.1 Experimental Setup

In order to experimentally identify favourable variants of variogram fitting, the methodological alternatives of experimental variogram aggregation (see Section 5.3.4) and fitting of the function parameters (see Section 5.3.5) are systematically tested.

The methodological parameters that are considered for systematic variation and testing are listed in Table 6.1.

Process	**Abbr.**	**Parameters/Variants**
Split Dimension Selection	split_dim	toggle, max_rel_dev, max_rel_ext
Split Position Selection	split_pos	mea, med, mid
Aggregation Position Selection	aggr_pos	mea, med, mid
Gauss-Newton Weighting Function	wgt_fnc	equ, lin, sin, log

Table 6.1: Methodological options for critical steps within the variogram fitting procedure

To apply and compare these variants in a simulation, a continuous random field is used. It was generated by applying a moving average filter on a field of pure white noise, as described in 5.3.1. The following properties have been applied to the generating process:

- grid size of 150 x 150 (spatial dimensions) x 30 (temporal dimension) elements (=675,000 grid cells)
- spatio-temporal extent in 150 m x 150 m x 60 min
- white noise field with mean of 5000 and deviation of 500
- filter: separable covariance function based on gaussian function for spatial and temporal dimension, spatial range of 50 grid cells ($\triangleq$ 50 m), temporal range of 15 grid cells ($\triangleq$ 30 min)

By transforming to greyscale levels, we get a visual impression of the model in Figure 6.7.

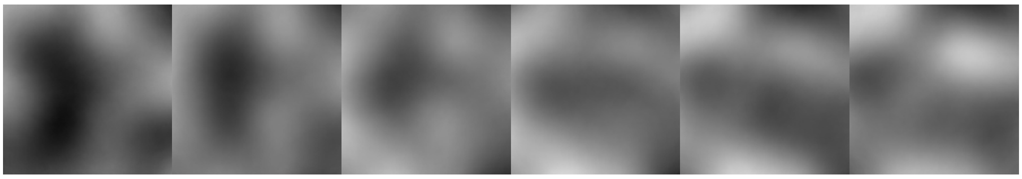


Figure 6.7: Experimental continuous random field as image sequence; images No 1, 4, 7, 10, 13, 16 out of the 30 image time series

According to Equation 5.8, we calculate

$$2\frac{150}{50} \cdot 2\frac{150}{50} \cdot 2\frac{60}{30} = 144 \qquad (6.5)$$

as the approximate number of observations necessary to capture the phenomenon adequately. The samples are dispersed randomly and uniformly over the set of 675,000 grid cells. For the experiments to follow, we keep the model and the random sampling positions constant to achieve identical conditions for all methodological variants.

By generating the experimental variogram, with equation 5.10 we get 23,220 pairings of observations which can be investigated in terms of a correlation pattern between spatio-temporal distance and squared halved difference of values (semivariance γ, see Equation 4.2).

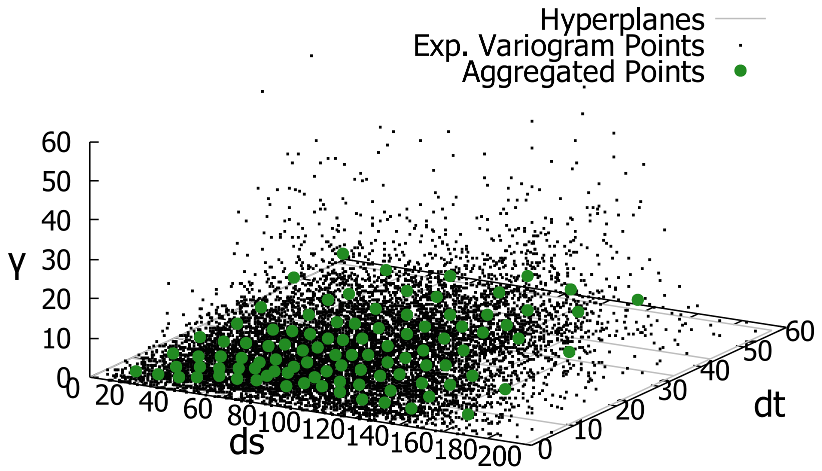


Figure 6.8: Variogram point cloud aggregation with semivariance γ for spatial (ds) and temporal (dt) distances; the hyperplanes are mainly concealed here and can better be seen in Figure 6.9

Figure 6.8 shows the semivariance γ of each pair on the vertical axis plotted against the spatial and temporal distance on the two horizontal axes, respectively. The aggregated green points are used to fit the theoretical variogram, as can be seen in Figure 6.9. The intersection lines of the partitioning BSP planes with the plane through $\gamma = 0$ are also plotted to illustrate the prior aggregation areas.

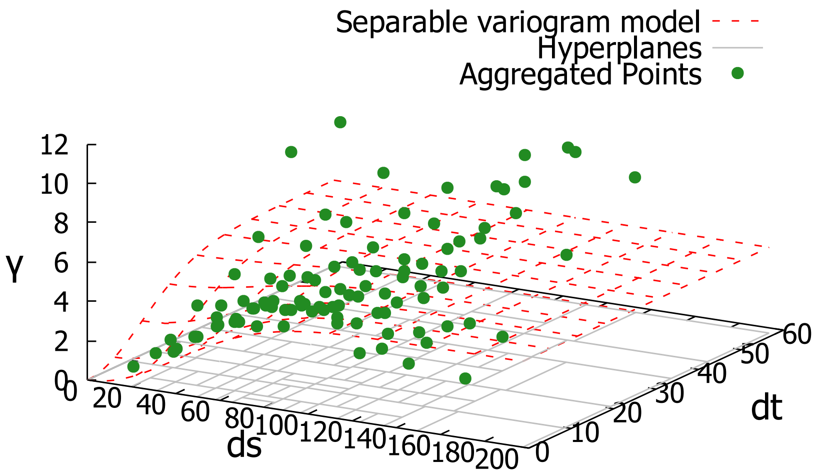


Figure 6.9: Separable variogram model fitted to aggregated points from experimental variogram point cloud with semivariance γ for spatial (ds) and temporal (dt) distances

Due to the scale of the vertical axis in Figure 6.9, the "outliers" in the regions of high spatio-temporal distance become visible. In variants where the weighting functions specified in Section 5.3.5 are applied, these points do not have much influence on the fitting.

To track down the particularly appropriate configurations from the methodological variants generally reasonable (see Table 6.1), all possible combinations of variants have to be iterated over. Since the order of process steps is invariant, the number of combinations is simply the product of variants per process step by

$$3 \cdot 3 \cdot 3 \cdot 4 = 108. \tag{6.6}$$

For each of these distinct parameter configurations, the RMSE is received by comparing the continuous random field with the interpolation result received by this particular configurational variant.

Since the estimation of the *range* values (spatial and temporal) is the crucial step within the whole process chain, these values are included in the evaluation scheme. More precisely, the sum of relative deviations of the estimated ranges (r_s, r_t) from the ones used for the random reference field (r_{sr}, r_{tr}) are used as metric for the quality of the overall variogram estimation by

$$d_r = \frac{\sqrt{{r_s}^2 + {r_t}^2}}{\sqrt{r_{sr}^2 + r_{tr}^2}}, \tag{6.7}$$

where small values for d_r indicate estimations of ranges near the ones used for random field generation by variogram filters (see Section 5.3.1).

To systematically compare all of the 108 parametric variants, the entire monitoring process chain is performed for each one of them.

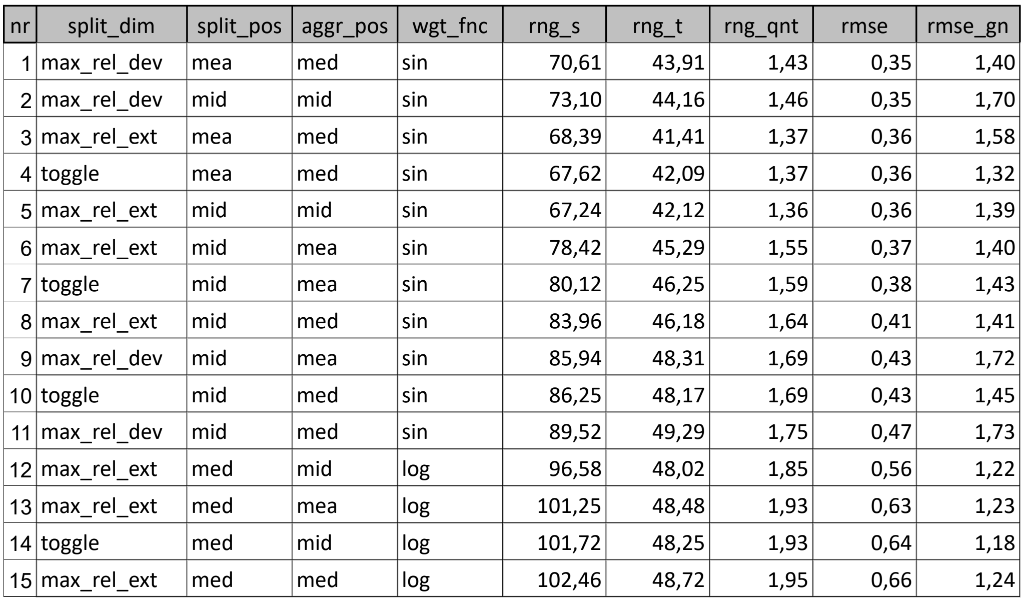
6.2.2 Results

The results of the experiments are presented as diagram in Figure 6.10. For the best 15 variants, a more detailed view is given in Table 6.2. It is sorted by ascending RMSE. The first four columns represent the parametric options with its selected values per row, whereas the remaining columns contain the numeric indicators considered significant for evaluation. The estimated range values for the spatial and temporal component of the variogram are listed as *rng_ s* *rng_s* and *rng_ t*, respectively. Derived from these, the combined quality estimation quotient calculated with Equation 6.7 is given by *rng_ qnt*. With *rmse_ gn*, also the residuals derived from the Gauss-Newton fitting procedure are considered.

For evaluation of all 108 parameter variants, the RMSE between reference model and interpolated model are plotted against two of the indicators described above as input variables *rmse_gn* and *rng_qnt* in Figure 6.10.

As can be seen from both plots, there is no obvious correlation between the RMSE from Gauss-Newton variogram fitting (*rmse_gn*) and the RMSE between reference model and interpolated model (*rmse*). In contrast to that, the quality of estimation of the (joint) *range rng_qnt* strongly correlates with the overall interpolation quality.

In both diagrams we find two clusters where variants have the same RMSE (2.0 and 3.4), which can only be seen in the left diagram where the abscissa values differ. The reason for that is that the corresponding methods for aggregation and fitting yield the same upper threshold for parameter estimation, which results in the same RMSE values. Since this upper parameter threshold



nr	split_dim	split_pos	aggr_pos	wgt_fnc	rng_s	rng_t	rng_qnt	rmse	rmse_gn
1	max_rel_dev	mea	med	sin	70,61	43,91	1,43	0,35	1,40
2	max_rel_dev	mid	mid	sin	73,10	44,16	1,46	0,35	1,70
3	max_rel_ext	mea	med	sin	68,39	41,41	1,37	0,36	1,58
4	toggle	mea	med	sin	67,62	42,09	1,37	0,36	1,32
5	max_rel_ext	mid	mid	sin	67,24	42,12	1,36	0,36	1,39
6	max_rel_ext	mid	mea	sin	78,42	45,29	1,55	0,37	1,40
7	toggle	mid	mea	sin	80,12	46,25	1,59	0,38	1,43
8	max_rel_ext	mid	med	sin	83,96	46,18	1,64	0,41	1,41
9	max_rel_dev	mid	mea	sin	85,94	48,31	1,69	0,43	1,72
10	toggle	mid	med	sin	86,25	48,17	1,69	0,43	1,45
11	max_rel_dev	mid	med	sin	89,52	49,29	1,75	0,47	1,73
12	max_rel_ext	med	mid	log	96,58	48,02	1,85	0,56	1,22
13	max_rel_ext	med	mea	log	101,25	48,48	1,93	0,63	1,23
14	toggle	med	mid	log	101,72	48,25	1,93	0,64	1,18
15	max_rel_ext	med	med	log	102,46	48,72	1,95	0,66	1,24

Table 6.2: Result table with systematic evaluation of 15 best out of 108 variogram aggregation variants sorted ascending by main quality indicator RMSE

is inadequate and produces weak results, this effect was not investigated any further.

The most significant property in this experimental series is the weighting function (wgt_fnc). The sine variant appears superior to all other functions, followed by the logarithm-based variant. The first variant that does not use different weights at all (*equ* for equal weights) appears at position 45 out of 108 and thus appears not to be beneficial in any constellation.

For the splitting position (*split_pos*) of the space partitioning algorithm the mean and middle positions perform best, while for the position to be aggregated (*aggr_pos*), the median and the middle appear beneficial, although not that significant, since the mean variant already appears at sixth position.

The least distinctive feature in this constellation is the method by which the next splitting dimension (*split_dim*) is determined, since all possible three variants are among the best four configurations. So the binary space partitioning algorithm appears to group the points into subsets in a way that is not significantly affected by this step according to the properties that are relevant for variogram generation.

Although the experiment does not in general indicate unambiguous advantages for particular variants of aggregation (except for the weighting function),

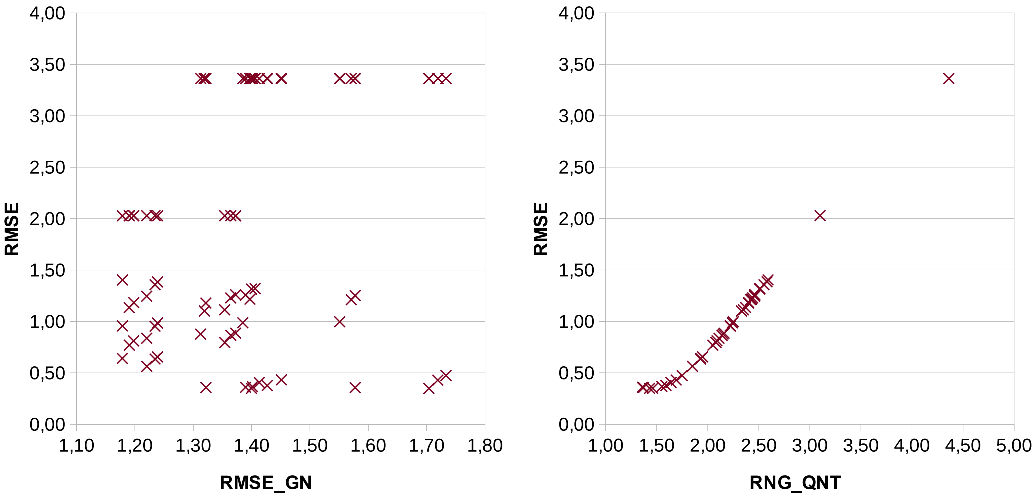


Figure 6.10: Evaluation diagrams of 108 parameter option variants; the RMSE between reference model and interpolation model (ordinate axis) is plotted against the RMSE from variogram fitting by Gauss-Newton (RMSE_GN, left) and the quotient between the compound range derived from variogram fitting and the one from model generation (RNG_QNT, right)

it certainly reveals some preferences that should be considered in forthcoming experiments.

For the most distinctive option *weighting function* there might be potential for further optimization by defining and testing variants similar to the sine or logarithm-based function. This strategy can also be applied to other—actual or future—options, thus evolving towards better and better solutions.

6.2.3 Conclusions

The monitoring of continuous phenomena is of high complexity, also because of the interdependencies of the variety of methods and parameters that can be applied [Meyers, 1997, p. 42 ff.]. The contribution of the variation module is to facilitate systematic tests and evaluations based on different indicators. The experiment introduced in this section was focused on the crucial task of geostatistics: the variogram estimation.

Although already numerous methodological variations have been evaluated, there is plenty left for further survey resulting from the possible variants of monitoring. If not only methodological options but also parametric values shall be varied (e.g. variable x from 3.0 to 8.0 in steps of 1.0), this can easily be included into the variation component.

With the proposed tool, more advanced analysis of the resulting evaluation tables like data mining or steepest ascent [Box and Draper, 2007, p. 188] are possible and might reveal more complex dependencies than the ones identified here.

It has to be stated that the experiment relies on one single synthetic model with a single set of observations. This is, however, a common situation in practice as [Matheron, 1988, p. 40]. However, the results presented here might very well be different for another synthetic model and also for another distribution of samples, so one should be reasonably reluctant from drawing general conclusions from them.

On the other hand, limited generality is the very nature of experimental studies and this does not mean that it is not possible to draw any conclusions at all from them. Rather, they might be considered valid until they are overridden by new experiments that provide deeper insight and more general laws [Popper, 2002], [Gigch, 1991, p. 62].

The general architecture of the simulation framework introduced here is supposed to make this process more efficient.

6.3 Sequential Merging

Sequential merging has been introduced in Section 5.4.2 as a technique to exploit the kriging variance in order to conflate several models by weighting their values by their inverse variance [ín Martínez and Sánchez-Meca, 2010]. The original motivation for this concept was to mitigate the computational burden of large sets of observations. In (near) real-time monitoring environments where the state model needs to be updated continuously by new observations, the problem of seamless merging can be solved by the same approach.

As a proof of concept, the sequential data merging method is tested experimentally and evaluated according to its performance gain in this section.

6.3.1 Experimental Setup

In order to prove the feasibility of the approach, but also to reveal its impact on accuracy, a simulation with appropriate indicators is performed. For this purpose, a synthetic continuous field is used. It is derived by kriging over 14 rain gauge stations and depicted in Figure 6.11 (a).

At this stage, we ignore temporal dynamism in order to keep it out as a factor for differences (RMSE) between the reference model and the sequential approach. In the simulation scenario, the continuous grid model serves as reference. Random observations are scattered over the model area, each assigned the value picked from the reference model at its position. Given this simulated measurement set, a new model can be calculated by kriging (Figure 6.11 (b)).

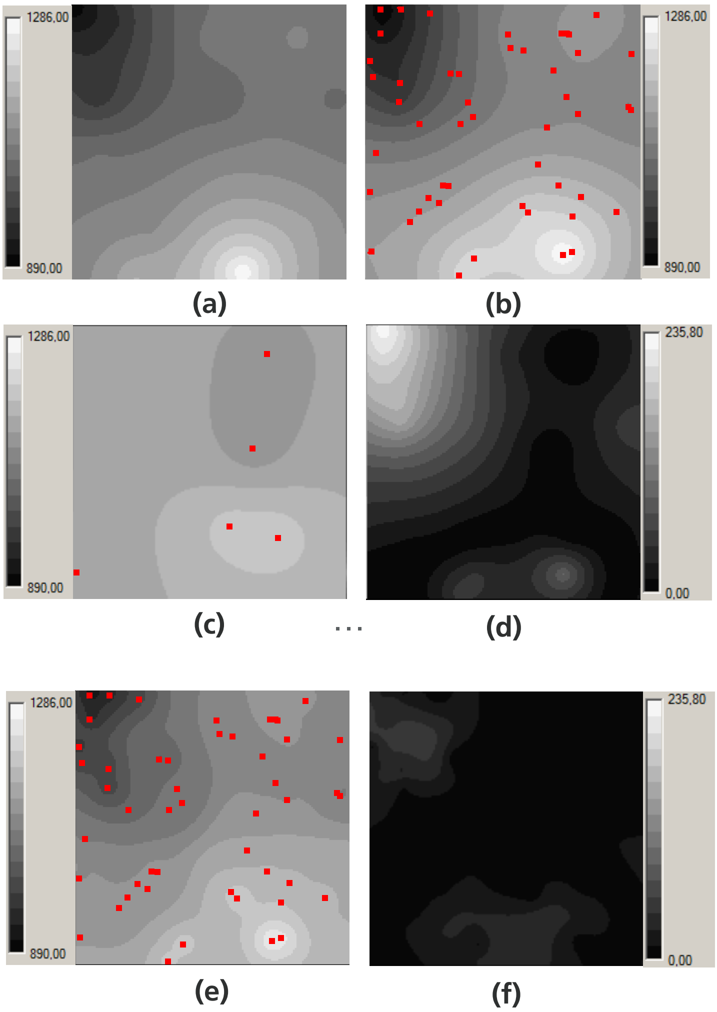


Figure 6.11: Evaluation of sequential method: reference model (a); random points and corresponding model derived from those points (b); first subset of random points (c) with corresponding difference map (d) against the reference (a); sequentially updated model of all subsets (e) with resulting difference map (f) against the reference (a).

The derived model (Figure 6.11 (b)) slightly differs from the reference model (Figure 6.11 (a)) due to interpolation uncertainty, but approximates it well when the number and distribution of samples are sufficient (see Section 5.3.2).

Following the sequential strategy, subsets of all synthetic measurements are created and calculated sequentially in sub-models (see Figure 5.15 and 5.16).

For the first subset (Figure 6.12 (c)), the deviations to the reference model (a) are rather large and can be seen in the difference map (d). Calculating all subsets of the data and merging them successively by weight leads to the final model (e), which also considered all the sample data, but unlike model (b) in a sequential manner.

The difference map (f) expresses the discrepancy towards the reference caused by the sequential approach. The overall discrepancy per model can be quantified by the root-mean-square error (RMSE) relative to the reference model (a). In the following, this value is used to indicate the fidelity of these interim models.

6.3.2 Results

In Figure 6.12, the computing time is plotted against the RMSE relative to the reference model for both the complete model calculation (square) and the sequential method (connected dots). Randomized sets of points (100, 200, 300 and 400) were subdivided into subsets or sub-models of 10 points each.

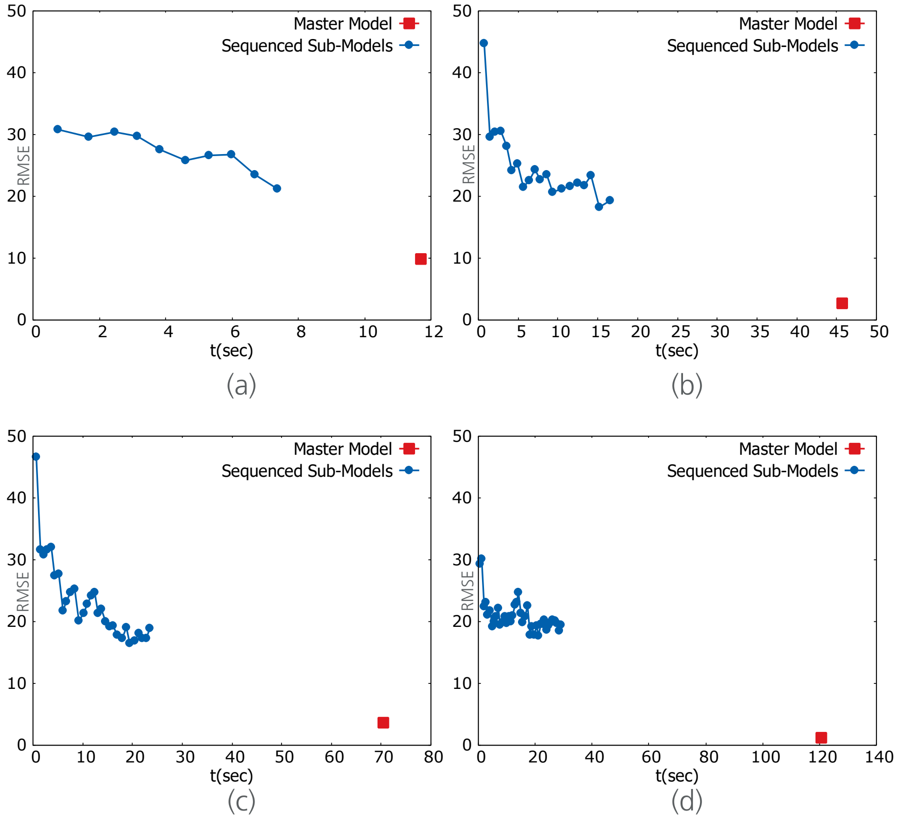


Figure 6.12: Performance comparison between master model (square) and sequenced calculation (dots): (a) 100 samples, (b) 200 samples, (c) 300 samples (d) 400 samples; subdivision is done in a way that sub-models contain 10 samples each.

As can be seen from the results, the sequential method has a lower accuracy in total, but provides a coarse result almost immediately. Within each plotted scenario, the RMSE tends to decrease when following the sequence. The $\mathcal{O}(n^3)$-effect of the conventional calculation becomes obvious when comparing its total computing time to the one of the sequential approach for large n.

Since the observations are distributed randomly over the reference model, the results also tend to scatter when the scenario calculation is repeated. But the general behaviour of the algorithm is reproducible in essence.

The tests introduced here are designed to explore the general behaviour of the approach. It converges to a saturation value and for large models clearly outperforms the conventional method in computing time.

6.3.3 Conclusions

This experiment was designed to demonstrate the general feasibility of a sequential strategy when performing kriging interpolation. It exploits the kriging variance as a continuous weighting schema of the models to be merged. The approach addresses the computational complexity of kriging for large datasets and the problem of integrating new observations into an existing model, as present in real-time monitoring scenarios.

The results show that the approach reduces total computing time for large datasets and provides coarse models immediately. It defines a rule for seamless merging of partial models based on the information confidence given by the variance map. For real-time monitoring systems that are fed by a continuous data stream, the method provides fast responsiveness and can adapt to data load and available resources.

For real monitoring scenarios, the method will have to be refined to generate acceptable results under given circumstances like data stream characteristics, model update intervals, computational resources and quality requirements. With the framework proposed, such circumstances can be considered in simulation scenarios. Different method variants and parameters can be applied and evaluated by using appropriate output indicators. Feasible approaches and settings can thus be identified before using them for real monitoring scenarios.

6.4 Compression

In this section, the compression algorithm as described in Section 5.4.3 is applied to empirical data. After an introduction of the specific data format of drifting buoy data, an evaluation of the achieved compression rate is carried out.

6.4.1 Experimental Setup

For the evaluation of the compression algorithm, data from the *Argo* drifting buoys program were used[1]. The format was provided by a Canadian governmental service[2]. It is used for the experiments and is described in Listing 6.1.

```
Contents:
Col 1 = Platform identifier (ARGOS #)
Col 2 = EXP$ - The originator's experiment number
Col 3 = WMO$ - WMO platform identifier number
Col 4 = Position year/month/day hour:minute (UTC)
Col 5 = Latitude of observation (+ve North)
Col 6 = Longitude of observation
        (+/- 180 deg +ve West of Greenwich)
Col 7 = Observation year/month/day hour:minute (UTC)
Col 8 = SSTP - Sea surface temperature (deg. C)
Col 9 = Drogue on/off - 1 = attached; 0 = not

Note: Missing value indicated by 999.9999
```

Listing 6.1: Original header of ARGO drifting buoy data

The sample contains all data types mentioned in Section 5.4.3. We find type *Integer* for the IDs in columns 1, 2 and 3. Colums 4 and 7 contain *DateTime* types, colums 5, 6 and 8 represent *Double* numbers while column 9 displays an on/off state as *Binary*.

From the original dataset, subsets of 100, 1000 and 10000 points are selected by spatial and temporal bounds. Listing 6.2 depicts a corresponding data header generated by the compression algorithm (note the changed names and order compared to Listing 6.1). The values for *min* and *max* are derived from the actual data. Together with the preset value *max_dev* for the maximum

[1] http://www.argo.ucsd.edu, visited 2018-02-19

[2] http://www.meds-sdmm.dfo-mpo.gc.ca/isdm-gdsi/drib-bder/svp-vcs/index-eng.asp, visited 2016-04-27

deviation, the bit depth is determined using Equation 5.21. The value *bpr* indicates the number of *bits per row* used for each column.

A maximum deviation of 0.5 for integer numbers means that at full bit depth the exact number is provided. For the *DateTime*-type this value represents seconds, so the minutes are decoded accurately when it is set to 30.

```
fname max_dev bits bpr min              max
x     0,0005  16   1   10,767           49,671
y     0,0005  15   1   40,07            59,08
val   0,0005  14   1   5,529            18,55
idarg 0,5     16   1   37411            92885
idexp 0,5     12   1   6129             9435
idwmo 0,5     23   1   1300518          6200926
tpos  30      10   1   2010-12-31 21:54 2011-01-01 09:24
tobs  30      10   1   2011-01-01 00:05 2011-01-01 09:57
drg   0       1    1   False            True
```

Listing 6.2: Header for the compressed dataset of ARGOS drifting buoy observations

As can be seen from Listing 6.2, the value for *idwmo* has the highest bit depth of 23, since the range of that value is nearly five million. The effect is that the longest chain of bits occurs for that dimension in the corresponding data file (see Listing 6.3).

```
   iii          iii          iii
   dddtt        dddtt        dddtt
  vaewpod      vaewpod      vaewpod
  arxmobr      arxmobr      arxmobr
xylgpossg    xylgpossg    xylgpossg

101011001    010101000    111101000
10100000     01110000     00010000
01001100     00100100     00000100
11000010     00010010     00010010
01001000     01001000     00101000
01100000     01111000     01011001
01101000     01001001     00101000
10000100     01101100     11101101
01000110     11000111     10000111
11100100     00010100     11110100
001001       101011       111011
000001       010111       011111
0100 0       0101 0       1010 0
1010 1       0111 1       0001 1
01 1 1       10 0 1       10 1 1
1  0 0       1  0 1       0  1 0
     1            0            1
     1            0            0
     1            1            0
     1            0            0
     1            0            0
     1            0            0
     0            1            1
```

Listing 6.3: Compressed data for three observations of ARGO drifting buoys (column names and values to be read vertically)

Three observations are listed, each containing all nine data columns organized vertically (as are the column names) with increasing accuracy from top to bottom. As can be seen, the binary value of the rightmost field *drg* (indicating drogue on/off) is already complete in the first row whereas the one for *idwmo* is resolved in row 23, as indicated in the header file (Listing 6.2).

Since this column represents an ID, it might very likely be necessary to resolve it earlier than in the last data row. Therefore, the number of bits per row is increased to four. The resulting structure for the same data can be seen in Listing 6.4.

```
   iii           iii           iii
   ddd   tt      ddd   tt      ddd   tt
  vaew   pod    vaew   pod    vaew   pod
  arxm   obr    arxm   obr    arxm   obr
xylgpo   ssg  xylgpo   ssg  xylgpo   ssg
------------  ------------  ------------
100011010001  010101010000  101011010000
10000000100   01000000100   11100000100
01001111100   00010111100   00101111100
01000011000   00010011100   00100011000
11001111100   01011001000   01101010110
11000110 00   01011001 00   11100100 00
11101    10   01011    10   11101    10
10000    10   11111    10   01100    00
11110    00   11010    00   10100    11
11000    10   10100    11   01100    11
00010    10   10011    11   01100    11
10110    00   01101    01   00100    01
1111          1110          0010
0010          1101          1110
0110          0001          1110
1 10          1 11          1 10
  0             0             1
  1             1             1
  1             1             1
  0             1             1
```

Listing 6.4: Compressed data with prolonged bit length of four per row for column *idwmo* (column names and values to be read vertically; data columns without name belong to *idwmo*)

In this configuration, the exact values for *idwmo* are already resolved in the sixth row since the three columns to the right without title are also utilized. In practice, the bit length per row can either be set directly (column *bpr* in the header), determined by maximum number of rows, or by some arbitrary combination of accuracy and row number in the form “accuracy x must be met in row n”. This configuration can be set individually for each dimension to achieve a good balance between stepwise accuracy improvement and total size per data row.

6.4.2 Results

To create indicators for the performance of the compression method, it is applied to a dataset of 100, 1,000 and 10,000 observations given in the format described above. We compare four indicators here: The first indicator is the size of the text file as received from the Canadian governmental service provider (denoted by “Text” in the following).

The necessary space when the data is parsed and translated into native machine data types is evaluated as second indicator (“Native”). We assume 32 bits for *Integer*, 64 bits for *Double*, 64 bits for *DateTime* and 8 bits for *Boolean*. The proposed binary format of the BSP compression algorithm is the third format listed. The size of the header is not considered here. Finally, a ZIP compression of the text file is applied as forth format with 7-Zip using following settings: normal compression level, deflate method, 32 KB dictionary size and a word size of 32.

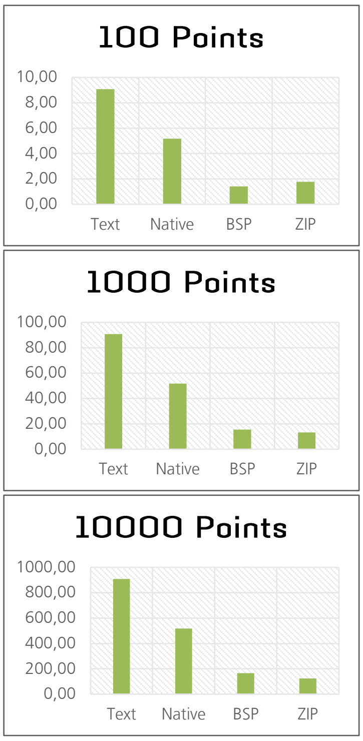


Figure 6.13: Data volumes (KB) in different formats for 3 datasets

As can be seen in Figure 6.13, the approach outperforms the ZIP compression for the small sample. With growing data size, the efficiency of the ZIP dictionary is increasing, which is not the case for our approach. Nevertheless, taking into account progressive decoding as an important key feature, the slightly worse compression ratio for large datasets appears acceptable.

Remarks on Reasonable Extensions The proposed compression method as introduced so far fulfils the requirements mentioned at the beginning of this section. There are, however, some ideas not yet implemented but certainly worth considering to be realized in future.

In the sample buoy data introduced in Section 6.4 we find missing measurement values indicated by "999.9999" (see header in Listing 6.1). The idea behind this number is to have an optical pattern immediately recognizable for the human eye as exception. Using it as "unset"-indicator within the compression algorithm is rather awkward, since, by being an absolute outlier, it enlarges the value domain (and therefore the necessary bit depth) significantly. A more explicit variant is desirable here, e.g. by indicating validity/invalidity of a value by its first bit. In case of invalidity, the bits to follow for that particular dimension can simply be dropped, but on the other hand, this mechanism would only pay off when having a significant amount of unset values.

Another aspect worth considering is the compression of the header associated with each compressed data segment (see Listing 6.2). With its metadata for each value dimension (name, deviation, bit depth, bits per row, min, max) it is crucial for archiving and retrieval and a prerequisite for correct decoding. In a monitoring scenario with very small data segments to be compressed, the relative size of that header can justify its compression where transmission of data is expensive.

Wherever the transmission of the compressed data is potentially error prone and not secured by other protocols, it might become necessary to implement some checksum method within the binary format itself. In this case, the gain in reliability needs to be carefully weighted against the expenses according to implementation, processing and data volume.

6.4.3 Conclusions

The methodology presented here is useful for situations where massive sensor data need to be compressed in a way that allows a progressive retrieval with increasing accuracy per step. It supports the most typical data types found in sensor data like *Float/Double*, *Integer*, *Boolean*, and *DateTime*, each one with specific compression schemata. The compression ratio depends on the value range and necessary accuracy. The number of bits per transmission step can be set in accordance with the transmission priorities, e.g. if certain dimensions are needed with higher convergence of accuracy per step.

In a wireless network scenario, the method would require some overhead for communication between data nodes. For example, the header that determines the mode of transmission needs to be exchanged before transferring the data. In environments where transmission of data is significantly more expensive than processing coding and decoding tasks, the method is likely to pay off.

For using the proposed method in a real-time environment, some protocol needs to be created to retain efficiency of transmission: values of defect sensors can be omitted, changed value ranges need to be adjusted and maybe the bits-per-row configuration shall be changed due to changed priorities. All this means considerable overhead which should carefully be weighed against achievable savings for data transmission.

When thinking about long-term archiving of data streams in databases, there are several points to be considered. Maybe the most important one is how a large dataset is to be segmented into smaller units. Doing this by spatial, temporal or spatio-temporal boundaries is reasonable since this is the most obvious means to refer the sensor data to other aspects like e.g. traffic density. Databases today widely support efficient management of spatial and spatio-temporal data [Brinkhoff, 2013].

But the associated indexing techniques were primarily developed having retrieval performance and not compression in mind. Thus, it appears reasonable to make use of them at a higher granularity level than the individual observation. So the method proposed here can be applied to appropriate segments of data while using the spatial or spatio-temporal boundaries of that segments for indexing with common database techniques. The compressed segment can be stored as binary large objects (BLOBs) in the database with associated spatial/spatio-temporal index and metadata.

Since the spatio-temporal boundaries can also be seen as statistical properties of the dataset, it is reasonable to ask if additional statistical properties like mean value, standard deviation or skewness should not also be considered for each dataset. This might be of little use for the dimensions space and time, but can be crucial for measured values like temperature or air pollutants. If advanced analysis methods like geostatistics are used, more complex statistical indicators like variogram model parameters should be considered [Lorkowski and Brinkhoff, 2015a]. All those data should be stored as metadata alongside with each dataset to support efficient retrieval.

One central issue here is the way large datasets are subdivided into smaller subsets on which the compression method is applied to and the corresponding metadata are related to. A good configuration balances retrieval granularity, subset management overhead, indexing costs, transmission data volume, system responsiveness and accuracy in a way that fulfils the requirements of the whole monitoring system.

6.5 Prediction of Computational Effort

The general idea behind the experiment set out in this section is to compare the computational effort predicted by the model approach of machine-independent description from Section 5.5.2 with the one actually measured in experiments. Different hardware with different configurations according to multithreading are applied to test the generality of the concept.

6.5.1 Experimental Setup

To evaluate the concept from Section 5.5.2, it was applied to the resource intensive process for generating continuous random fields. The computational workload of this algorithm, obtained with help of the function *QueryProcessCycleTime* and expressed by the metric *gigacycles*, increases when increasing the *range* value of the associated variogram, since more grid cells have to be considered to calculate the weighted mean.

The experiments were carried out for different modes on two different CPUs: an Intel® Core™ i3-5010U CPU with 2.1 GHz and 4 logical processors (2 cores) and an Intel® Xeon™ E5-2690 v3 with 2.6 GHz and 24 logical processors (12 cores).

In the study, the multithreading overhead factor *thro* from Equation 5.22 was quantified to 0.5 for both processors, which means that, in this case, the gain in performance in effect coincides with the number of *physical* (not *logical*) processors of the used CPU. This indicates that the multithreading functionality within one core can rarely be exploited for this task. For the Xeon™ CPU, the frequency correction factor fc is set to 1.5 to express its apparently better instructions-cycle ratio.

By switching the parallelization mode on and off for the critical loop in the algorithm, we get four configurations to evaluate according to the proposed model for estimation of computational effort. The cycles counted by the Intel™ Core™ i3-5010U CPU in the singlethreaded mode are used as reference for Equation 5.22.

In the given process, there is one portion of code that can only be processed singlethreaded because it contains routines difficult to parallelize. The other portion contains the critical loop executing the moving average filter by numerous iterations. Because of its high workload impact, this loop has deliberately been optimised with respect to parallelization.

6.5.2 Results

Based on the proposed metric, Figure 6.14 refers the time expense predicted by the equation (lines) to the time actually needed for calculation (points). The *calculated* time effort represented by the lines is composed of the sum of algorithmic portions for each workload position on the abscissa.

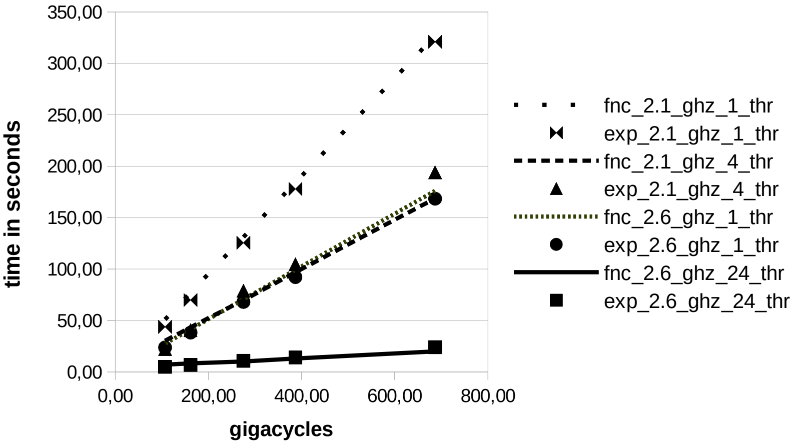


Figure 6.14: Performance evaluation of four computer system configurations: (1) 2.1 GHz with 1 thread, (2) 2.1 GHz with 4 threads, (3) 2.6 GHz with 1 thread and (4) 2.6 GHz with 24 threads. The lines (prefix *fnc*) represent the prediction by *function* for each configuration and the points (prefix *exp*) represent the *experimental* data.

The plot clearly reveals the scaling effects of multithreaded processing, which, as already stated in Section 5.5.2, is the crucial leverage for contemporary performance improvement. It can also be seen that predicted and actual time expenses have similar values. Only the variant with 2.1 GHz and 4 threads does not scale as well as predicted. One or some of the blurring influences mentioned in Section 5.5.2 can be assumed to cause this effect.

6.5.3 Conclusions

The experiment was focused on the generic quantification of computational workload in order to estimate the temporal effort that is necessary on different platforms. The evaluated toolset is capable of estimating the processing time of complex calculations for different configurations of scenarios of monitoring, analysis and simulation.

In combination with the toolset for systematic variation and evaluation (see Section 5.5), this approach allows for deep analysis of multiple constellations according to methods, parameters and hardware configurations and their effects on performance indicators. Feasibility and efficiency studies for different configurations can thus be carried out without actually using the intended hardware. A calibration of the parameters of Equations 5.22 and 5.23 might be necessary if they can not sufficiently be obtained from hardware specifications.

The central concern of the concept is to abstract from concrete hardware configurations, algorithm parameterizations and output indicators and thus to simplify and standardize comprehensive evaluation scenarios. Strategies for incremental optimization are thus made explicit and transparent. This can significantly help to make experimental results well documented and reproducible.

The modelling and consideration of data transmission costs, which is crucial for wireless networks, was only briefly mentioned as one possible factor of optimization. An inspiring scenario in this context would be a complex simulation with distributed sensors and full knowledge about geometric constellation and profiles for data transmission expense that depend on that constellation. Placing the sensors on autonomous platforms increases complexity and imposes extensive simulation to find strategies for a high overall efficiency. The tools presented here could be of significant importance to master the complexity of such missions.

6.6 Case Study: Satellite Temperature Data

The experiments introduced so far were carried out on synthetic models generated by a filter (see Section 5.3.1) or on models derived from interpolated observations (as in Section 5.4.2). The idea behind this approach is to have full knowledge about the observed phenomenon.

Beside this, working with synthetically generated models has the advantage that the variogram parameters estimated from the observations (see Section 5.3.5) can be compared to the ones known a priori. Interdependencies between the quality of parameter estimation and overall interpolation quality can thus be identified (see Section 6.2). Furthermore, the formula for the minimum sampling density (Equation 5.8) can thus be tested for different constellations of dynamism.

To also apply the framework to *empirical* data, a remote sensing thermal image, obtained from the National Oceanic and Atmospheric Administration (NOAA), is used as reference in this section. In practice, such imagery data usually does not have to be interpolated since it already contains the variable of interest in the required resolution (except for occlusions, e.g. by clouds [Sun et al., 2006, Cressie and Wikle, 2011]). So the advantage of knowing the complete model in the given resolution is also given here. What is *not* given here—in contrast to the synthetically produced model—is any a priori information about the dynamism of the observed phenomenon.

6.6.1 Experimental Setup

Just as with the synthetic data, in this experiment the given raster image is also sampled with random observations that are used to estimate the unobserved grid cells. Analogously, the accuracy of the interpolation can then be quantified by the difference between the reference and the derived model.

As reference model, a satellite raster image of the *4km Pathfinder SST Climatology* provided by the National Centers for Environmental Information (NCEI)[3] from the National Oceanic and Atmospheric Administration (NOAA), was selected. It provides the sea surface temperature with a ground resolution of about 4.6 kilometers (for both latitude and longitude, since a region in the Atlantic Ocean near the equator at $4°S, 12°W$ was chosen). The image entails 150 rows and 150 columns, resulting in 22,500 grid cells. An area of about 697 * 697 km is covered. The image was taken at night on 2013-01-01. The temperature is stored as 32 bit floating point value of unit kelvin.

Figure 6.15 displays the grid using grey scaled values.

[3] `https://www.ncdc.noaa.gov/cdr/oceanic/sea-surface-temperature-pathfinder`, visited 2018-02-19

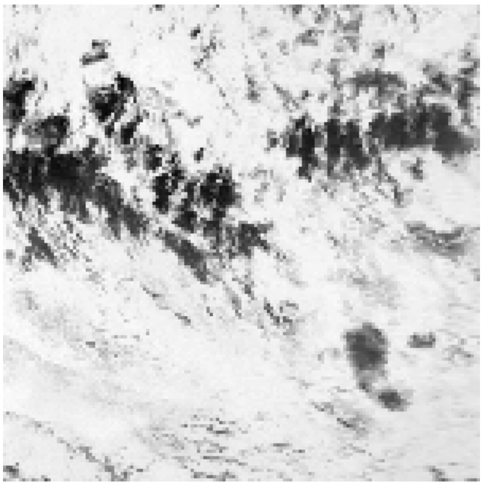


Figure 6.15: Sea surface temperature (SST) satellite image extracted from the 4km Pathfinder SST Climatology provided by the National Centers for Environmental Information (NCEI). The values range from 284.0 (bright) to 298.9 (dark) K (10.8 to 25.7 $°C$, respectively)

Assuming a sufficient signal-to-noise ratio, the grid represents the sea surface temperature (SST) at the given resolution. Although a predominantly continuous character can be granted, there are also discontinuous patterns, especially in the upper half of the image. While the overall standard deviation of the temperature grid is 3.3 K, we find differences of more than 10 K for neighbouring cells at the edges of these patterns.

These effects are caused by ocean circulation and oceanic fronts [Vihma et al., 2014, Sabins, 1996]. This issue is extensively covered by explicitly determining such fronts in an interpolation model in [Sun et al., 2006]. The method is proposed for applications where discontinuities are rather common, like for oceanography or soil moisture monitoring. Gaps in observational coverage, e.g. caused by clouds, are thus bridged by patterns that entail such fronts.

The subject of discontinuities is beyond the scope of this work. Instead, the set of methodological and parametrical variations given by the framework are applied to the image in order to identify the best configuration from the given set of variants for this particular kind of phenomenon.

The satellite image indicates the superiority of the remote sensing method. In this example, it provides a gapless and consistent representation of the phenomenon. Nevertheless, remote sensing is not always available due to clouding, coverage, cost, etc. [Sabins, 1996]. Moreover, not every phenomenon can be acquired sufficiently by imaging techniques like remote sensing (see also Section 2.1).

Consequently, discrete sensor observations, although maybe very sparse, are often the only source of knowledge that can actually be obtained. Even if the estimation of a value by kriging interpolation between those sparse observations might not be precise at all, it often provides the best results—according to bias and variance—that can be generated from these observations [Cressie, 1990, p. 239], [Oliver and Webster, 2015, p. 88].

Notwithstanding the presence of partial discontinuities, the necessary sampling density for the image was estimated by Equation 5.8. Because it entails the *range* parameter, which is not known beforehand for this dataset, it was estimated iteratively to be about 22 grid cells. With Equation 5.8, we get 186 observations as minimum density, which was rounded to 200 randomly dispersed observations for the experiment. This results in 19,900 variogram points (see Section 4.2) by pairwise combination.

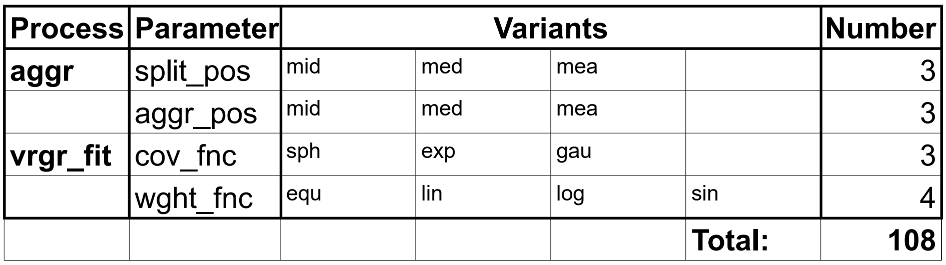
It has to be stated, though, that the equation was deduced for phenomena that are considered stationary, which is not strictly the case for the given dataset. Depending on the aim of the interpolation, this approach might nevertheless very well be reasonable, as will be discussed with respect to the results below.

As already mentioned, the idea of the experiments described in this section is to treat the grid derived from satellite observation as reference, just like the synthetic random fields are treated in Section 5.3.1. Analogously, a random set of cells is used as observations and error assessment is also performed by calculating the difference between the reference model—in this case the satellite image—and the one derived from the interpolated observations. Consequently, the error assessment can be used to evaluate the quality of the chosen interpolation method as a whole.

In analogy to the variation of methods and parameters (Section 5.5) carried out for finding the best variogram fitting configuration (Section 6.2.1), a list of options is set up in Table 6.3.

Again, the options the table contains are systematically combined to cover all possible configurations. In addition to the parameters used in Section 6.2.1, the type of the covariance function *cov_fnc* was also chosen to be varied in order to allow for better adaptation to the specifics of the given image. On the other hand, the parameter *split_dim* for selecting the split dimension is not necessary here because with a purely spatial variogram there is only *one* dimension the splitting hyperplane can be moved along. Therefore, together with the other options, the number of variants also sums up to 108.

Searching for the optimal configuration would be rather cumbersome without the tool introduced in Section 5.5.



Process	Parameter	Variants				Number
aggr	split_pos	mid	med	mea		3
	aggr_pos	mid	med	mea		3
vrgr_fit	cov_fnc	sph	exp	gau		3
	wght_fnc	equ	lin	log	sin	4
					Total:	**108**

Table 6.3: Process method variants for interpolation of sea surface temperature

6.6.2 Results

In analogy to the evaluation in Section 6.2.1, the results of the 108 simulations are assessed by plotting the primary quality indicator RMSE against two other indicators: the residuals from the Gauss-Newton fitting procedure ($RMSE\_GN$) and the *range* value (RNG).

As Figure 6.16 reveals, there is no strong correlation between *range* and RMSE as has been in the experiment of Section 6.2.1. Unlike in that experiment, we do not find such a distinct pattern of *range* for the satellite image. Phenomena like circulation cells and eddies [Sabins, 1996, Peng et al., 2001] partly disturb the continuity of the system and therefore also the strong correlation between estimated range and interpolation quality.

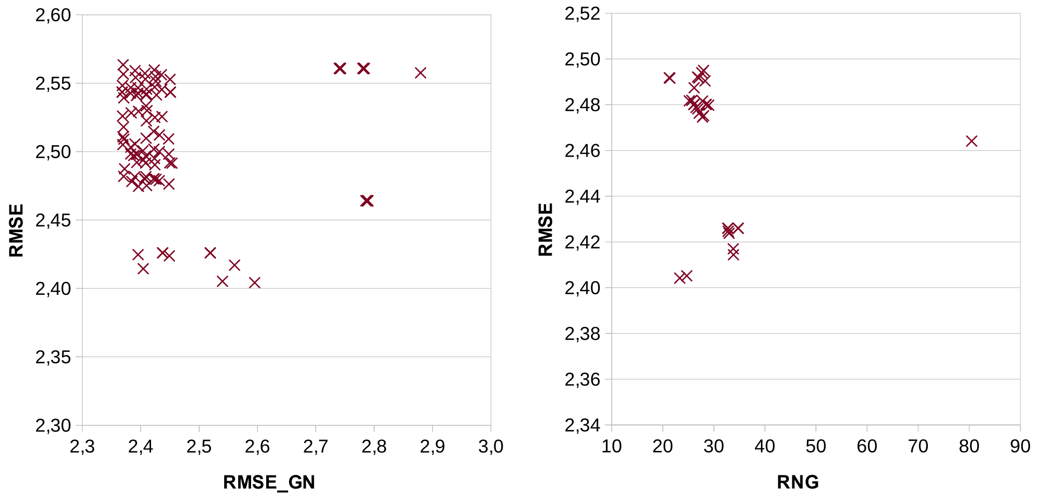


Figure 6.16: Evaluation diagrams of 108 parameter option variants with RMSE values plotted against the residuals from Gauss-Newton optimization $RMSE\_GN$ (l), and against the range RNG (r)

The 15 variants with lowest RMSE that are listed listed in Table 6.4 reveal the superiority of the exponential covariance function, the logarithm-based weighting function (*wgt_fnc*) and the median value for both partitioning parameters *split position* (*split_pos*) and *aggregation position* (*aggr_pos*).

nr	split_pos	aggr_pos	wgt_fnc	cov_fnc	rng	rmse	rmse_gn
1	med	med	log	exp	23,33	2,40	2,60
2	mea	mid	log	exp	24,70	2,41	2,54
3	med	mid	sin	exp	33,84	2,41	2,40
4	med	mid	lin	exp	33,80	2,42	2,56
5	mid	mid	log	exp	32,99	2,42	2,45
6	med	mid	log	exp	32,81	2,42	2,40
7	med	mea	log	exp	32,75	2,43	2,44
8	mid	mid	sin	exp	34,76	2,43	2,52
9	mid	mea	equ	exp	80,48	2,46	2,79
10	mid	mea	lin	exp	80,48	2,46	2,79
11	mid	mea	log	exp	80,48	2,46	2,79
12	mid	mea	sin	exp	80,48	2,46	2,79
13	mid	med	equ	exp	80,48	2,46	2,79
14	mid	med	lin	exp	80,48	2,46	2,79
15	mid	med	log	exp	80,48	2,46	2,79

Table 6.4: Listing of the 15 of 108 configuration variants with the lowest RMSE

Figure 6.17 illustrates the fitting of the variogram model to the aggregated points from the experimental variogram that yields the smallest RMSE when applied for interpolation (first row of Table 6.4).

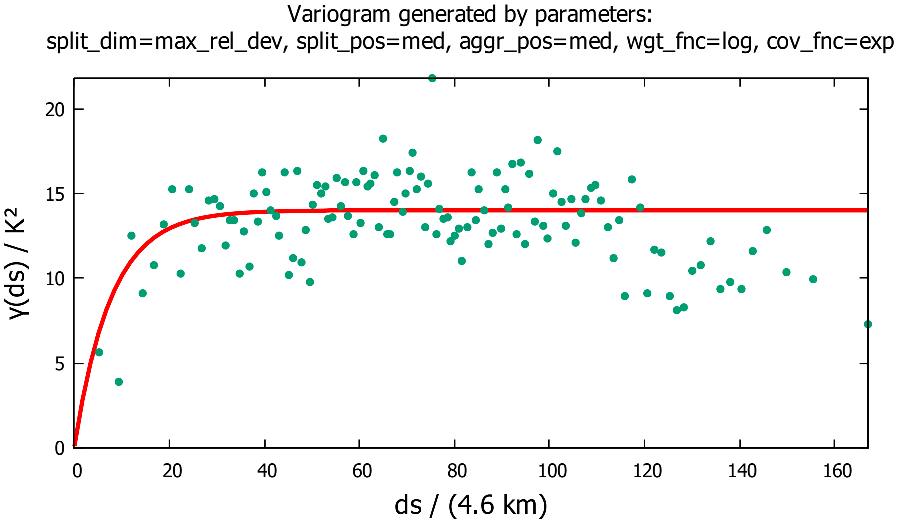
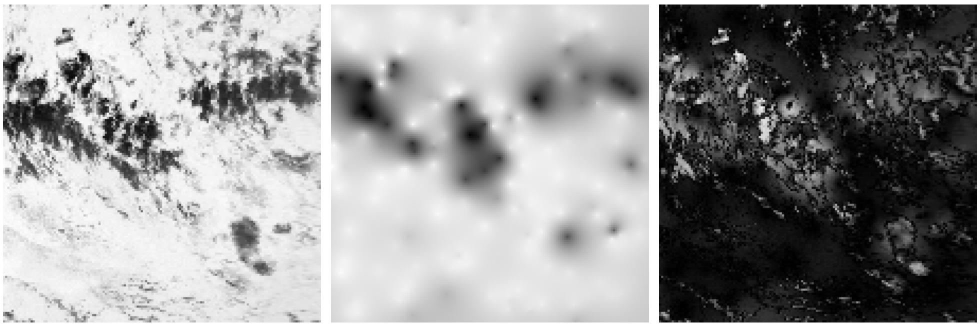


Figure 6.17: Variogram generated by the random observations on the sea surface temperature (SST) image; the distance unit 4.6 km results from the pixelwise treatment of the data, the unit K^2 is due to the square expression within the variogram (Equation 4.2)

The point distribution reveals the striking pattern of a decreasing semivariance for big distances, which indicates the anomaly of non-stationarity caused by oceanic fronts, as already mentioned above. Apart from the discontinuities in the upper half of the satellite image (Figure 6.15), there are large areas of similar value that are distributed all over the region. So there is a considerable amount of rather distant pairs of observations with semivariances that are smaller than the average, which leads to the variogram pattern as can be found in Figure 6.17.



(a) Reference satellite image (see Figure 6.15 for source specification)

(b) Grid derived from interpolation of samples by kriging

(c) Difference map between reference image and interpolated grid

Figure 6.18: SST satellite image on which 200 random observations were carried out for interpolation by kriging; from that, a difference map can be derived

6.6.3 Conclusions

As the kriged field grid in Figure 6.18(b) shows, the interpolated model does not seem to accordingly represent the patterns that can be identified in the reference satellite image. There are also interpolation artefacts manifested as *prussian helmets* [Webster and Oliver, 2007, p. 39], indicating some degree of incompatibility between the observed phenomenon and the interpolated model. This discrepancy also induces the noticeable textured difference image (Figure 6.18(c)).

These negative quality indicators do not, however, disqualify the interpolation method as a whole. It can still provide a valid representation of the phenomenon in terms of minimum RMSE towards the reference, which was target indicator used here.

When an authentic realization according to the covariance structure underlying the phenomenon is prioritized, a conditional simulation [Burrough et al., 2015, p. 190] might be preferred. If the focus is to estimate the average temperature of a region—e.g. for energy calculations in hurricane models [Michaud, 2001]—a target indicator like the deviation of the mean should be considered in search of appropriate methods and parameters.

So the selection of the *appropriate interpolator* depends on the objectives and circumstances of the monitoring as much as on the phenomenon itself

[Peng et al., 2001, p. 160]. When observing a high resolution reference model and comparing the interpolated model with it, as was carried out in this experiment, the interpolation method can adapt to those conditions.

While following the paradigm of the "closed loop" [Sun and Sun, 2015, p. 9], the framework presented here allows for variation of reference models, interpolation methods, parameter options, and indicators for quality and efficiency in order to address various objectives.

Chapter 7

Conclusions and Perspective

The monitoring of continuous phenomena is a complex and challenging task on many levels. In this thesis, a holistic concept has been proposed to divide this task into small and cohesive units of operation. They have to be processed sequentially and are interdependent since each process step uses the output of its predecessor as input. There are unlimited options of configuration within this process chain to control its behaviour.

In order to automatize and standardize the process of continuous improvement, a generic model for systematic variation of methods and parameters has been worked out. For evaluation of these generated variants, diverse indicators have been identified and specified, of which model quality and the computational workload are the ones that were examined closer here.

The main contributions of this work to the subject area of monitoring continuous phenomena are summarized below:

Minimum Sampling Density Estimation (Sections 5.3.2, 6.1) In order to estimate the average sampling density that is sufficient to capture a particular phenomenon, a formula was deduced that derives this density from the extent and the dynamism parameter *range*. It presumes the *range* parameter to be known or to be derived from initial observations.

Variogram Fitting (Sections 5.3.5, 6.2) Variogram fitting is the key task of geostatistics. A new and generic method based on binary space partitioning (BSP) was proposed to aggregate variogram points of arbitrary dimensionality in order to unburden the subsequent parameter fitting procedure.

Model Merging (Sections 5.4.2, 6.3) The merging of several grid models of one spatial region addresses two challenges that are common for monitoring systems: (1) the continuous and smooth update of a real-time model by new observations and (2) the handling of large sets of observations by subdivision (divide-and-conquer approach). The method exploits the kriging variance to define reasonable weights for cells of the source models by which they are combined.

Compression of Sensor Data (Sections 5.4.3, 6.4) An algorithm for compression and progressive retrieval of observational data of arbitrary dimensionality was proposed. Its predominant aim is to improve the sufficiency of sensor data transmission and archiving. In progressive mode, it provides coarse values of low data volume and increases in accuracy with each transmission step.

Systematic Variation and Evaluation of Configurations (Section 5.5) The tools described above entail unlimited potential for variation and configuration in order to adapt to the observed phenomena and to thus provide optimal interpolation results. When combining the variations within a process chain, the number of possible configurations to test and evaluate might quickly multiply to large numbers. To handle this complexity within a simulation framework is then a problem in its own right. A generic and coherent architecture to switch between methodological variants or to iterate numerical parameters was designed.

Quantification of Computational Workload (Section 5.5.2) Limited computational resources are often a crucial issue, especially for wireless sensor networks, environments with real-time requirements, and large datasets. A strategy for machine-independent description of computational workloads was developed and tested. It keeps track of the number of CPU cycles while differentiating portions of code *capable* and *not capable* of multithreading. While the conventional quantification by execution time ignores the parallelization facilities of both software and hardware, the proposed approach provides much more sophisticated information about the scalability of an implementation.

The features above have been implemented to plan, perform, provide, archive, evaluate and continuously optimise environmental monitoring processes and their results. Most of the proposed algorithms have been tested and evaluated with support of the tool for systematic variation and evaluation. Beside using synthetic models as reference, the algorithms were also applied to real world data, namely a satellite image for sea surface temperature (see Section 6.6). The concept for quantification of computational workload was tested for the computing-intensive task of random field generation.

In the course of the work there was a constant evolution towards increasingly abstract concepts to describe and quantify the manifold aspects of environmental monitoring. This is the case for *input* parameters like phenomenon dynamism, sampling density, or possible transmission bandwidth. The *system* might be specified by its algorithms, parameters, workload, computer power, storage space, energy demand, and response time. The generated *output* will predominately be evaluated by its accuracy and resolution. For a thorough planning and/or evaluation of a monitoring system, a systematic consideration of these issues are certainly helpful (see Section 5.5.4).

The proposed solutions and evaluation tools do undoubtedly leave room for further investigation and improvement. The intention of the work was not to thoroughly investigate one narrow problem area, but to aim at several challenges that are specific for the monitoring of continuous phenomena. For systematic evaluation of the efficiency of various solutions, a generic framework is provided. New methods with associated parameter settings can easily be integrated and evaluated by using the present infrastructure. The circular arrangement of the process chain—with genuine RMSE between synthetic and derived model—allows for iterative investigation and improvement of the monitoring task as a whole.

There are yet many challenges to overcome on the way towards a comprehensive, generic and consistent solution for environmental monitoring. In the long-term perspective, one might envision a standardized and interoperable infrastructure that mediates between the available sensor observations and an adequate representation of the phenomenon. Given the insights and achievements of this work, the prerequisites for such an environment are listed below:

Interpolation Method Consolidation Depending on the phenomenon observed, the aims of the monitoring and the resources available to achieve it, the method of interpolation has to be chosen carefully. Because of the sheer complexity of the task of interpolation and the plethora of methods available to address it, it is in most cases difficult to decide which one serves the given objectives best. Much specific knowledge and experience is necessary to come to well-founded decisions here.

There are many works dealing with sensor observations and interpolation with different methods and/or parameters. Evaluation is then typically carried out by metrics generated by methods like cross-validation [Burrough et al., 2015, Gama and Pedersen, 2007, Meyers, 1997]. The problem with these approaches is that cross-validation is not always an indicator for interpolation quality or, pointed out in [Oliver and Webster, 2015, p. 68]: "The results of cross-validation do not necessarily resolve or justify a choice of model." Continuous random fields and simulation scenarios provide an alternative way of evaluation of methods and parameters.

There is, of course, the disadvantage that the validity of the synthetic models might be doubted. This does not mean, however, that it precludes to draw valid conclusions from such models that are useful in practice. This is the case because there is no absolute model validity anyway [Law, 2014, p. 247] and real environmental phenomena do only approximately represent random processes [Ginevan and Splitstone, 2004].

In view of the vast amount of interpolation methods and their variants and associated parameters, there ought to be some mapping policy that relates phenomenon characteristics to interpolation variants which best adapt to those particular characteristics. A simulation framework with specialized features for systematic variation of both the phenomenon and the interpolation method facilitates the necessary steps towards this goal. The present work might contribute some useful approaches to this endeavour.

Formal Method Specification As mentioned above, there is an ever-growing amount of interpolation methods and associated variants and parameters, and also of input and output data formats. The way these specifics are addressed will substantially depend on the particular system and the developers' view on the problem that is factually materialized in form of the implemented method calling convention.

These circumstances might make it difficult to reproduce a particular processing scenario when a different software product has to be used for some reason. Even more effort might be necessary when the interpolation task is supposed to be run using software as a service (SaaS) where eventually an established interface has to be changed.

One approach to address such problems would be to pursue some degree of interoperability by working out an abstract and standardized definition of the process of interpolation. Even if such a standard would not be directly implemented by software suppliers, it would at least provide a common ground for communication about how a particular implementation works instead of more or less taking it as a black box.

Such a common ground specification will of course presume the agreement about its necessity and general structure. The general trend towards cloud computing with its associated advantages might foster developments towards such an agreed specification.

Field Data Type for Data Provision Observations of continuous phenomena are of specific character depending on the circumstances under which they have to bee carried out. More often than not, they do not cover the points or regions that are of interest for the task or question at hand. So instead of the original observational data, applications rather need estimations of the observed variable at arbitrary points or regions in space and time. This is crucial to perform any analysis that is based on the combination of the observed value with some other occurrence, often in order to identify any causative interaction [Cressie and Wikle, 2011, p. 32].

To support such analyses or also for ad hoc queries in space and time, it is necessary to provide an infrastructure that can serve as mediator between raw data and expected queries. One important means to address this task is the introduction of a specific field data type that is intended to represent continuous phenomena [Liang et al., 2016, Camara et al., 2014, Couclelis, 1992].

Unfortunately, the described concept of a field data type presumes the attributes already mentioned: the consolidation and the standardised description of methodological variants of interpolation.

In an ideal scenario, such a system is continuously fed with observations and simultaneously provides a real-time model or historical data via a standardized interface. Apart from interpolated and cached data to speed up queries, the observational data could be stored without redundant interpolations. Thus, the model of the phenomenon can be provided dynamically at arbitrary points or regions as a function of original observations and the associated interpolation specifications. Such an infrastructure would significantly increase the usability of such observational data and therefore widen the range of their utilization.

Complex Event Definition Another feature that was mentioned but not thoroughly covered in this work is the definition of complex events that are associated with a continuous dynamic field. It entails the specification of a spatio-temporal region for which some condition about the observed value has to be confirmed or rejected. An exceeded daily maximum value of a particular pollutant in a city district would be an example. More complex aggregations like mean or variance or combinations of them—e.g. to permanently exclude hazardous concentrations by observations—might be more appropriate. The definition of such a hazardous situation as null hypothesis that permanently has to be rejected [Guttorp, 2001, p. 24] would entail insufficient observation, indicated by high regional kriging variance, as alarm triggering event.

Furthermore, the spatio-temporal regions under investigation might be more complex than static areas. A trajectory can also be used to interact with the model. For example, the radiation that will be accumulated by a vehicle during a planned mission in a contaminated region could be estimated by spatio-temporally intersecting the trajectory with the model and accumulating the radioactive exposure.

So when given an appropriate model, the continuous phenomenon just means that—in a figurative sense—it is possible to place or move sensors arbitrarily in the region that is sufficiently covered by interpolation. This kind of continuous replication of the observed field is a powerful approach wherever flexible usage of the variable of interest is needed. Other representations like grids or isolines are not that flexible. However, they can of course easily be derived from the field data type at arbitrary resolution.

From the features listed above, each single one is more or less dependent on its predecessors in the given order. Although these features are interdependent in operation, the yet unsolved problems which they contain can—at least to a certain degree—maybe be demarcated through abstraction and thus be worked on individually. This provides plenty material for challenging scientific work in this area.

The circular principle of simulated monitoring presented in this work can easily be extended to develop and evaluate solutions associated with the scenarios described above. The detection of a critical state in a continuous field may serve as an example. The critical state can be expressed by a polygon for which a maximum daily average of the value of interest is defined. The actual average value can be determined from the three-dimensional grid (x, y, t) interpolated from the simulated observations by aggregating all grid cells within the spatio-temporal region of interest (polygon and day). By considering the kriging variance when performing such aggregations, a confidence interval for the estimated value can be deduced.

This scenario is an example of how knowledge about a continuous phenomenon is expressed on a higher level of abstraction. Instead of dealing with individual sensor observations, the context here are aggregations and probabilities or confidence estimations. From the current developments in this subject area it can be concluded that for responsive monitoring systems such an abstraction will become more common. The process of interpolation itself will further diversify according to methods and variants and a main challenge here will be to assess the dynamism of a phenomenon by observations and consequently choose the appropriate interpolation method with appropriate associated parameters. Extensive experimental work will help to find and formulate general rules to govern this decision process.

On the input side of monitoring, there has been much research on behalf of network organization and data transmission strategies. The efficiency of this component and the subsequent data processing will remain subject to continuing investigation and improvement. On the output side, an increasing degree of abstraction with respect to the representation of knowledge about continuous phenomena becomes apparent when regarding concepts like the field data type or aggregations derived from it.

Future developments might integrate data stream management, selection and execution of appropriate interpolation algorithms and parameters, sensor data data management with help of the field data type, provision of a query language suited for the context of continuous phenomena and, based on that, an infrastructure for critical state detection and notification.

To address this problem field in an appropriate and substantial manner, several conflicting requirements need to be carefully balanced: capability, performance, efficiency, interoperability, extensibility, ease of use, credibility and popularity. The priorities of these requirements will change several times during a system's life cycle.

With the increasing availability of environmental observations and the growing demand for actual knowledge derived from them [Craglia et al., 2012], it is just a question of consequential reasoning to come to a compound of modular solutions as suggested in this work. Just like with other subject areas in the realm of geographic information science, the approaches for the monitoring of continuous phenomena will have to undergo a continuous process of consolidation and specification before becoming a ubiquitous standard.

References

[Abrahamsen, 1997] Abrahamsen, P. (1997). *A review of Gaussian random fields and correlation functions.* Norsk Regnesentral/Norwegian Computing Center.

[Agterberg, 1974] Agterberg, F. P. (1974). *Geomathematics: Mathematical Background and Geo-science Applications (Development in Geomathematics).* Elsevier Science Ltd.

[Ahmed et al., 2012] Ahmed, M. H., Dobre, O., and Almatarneh, R. (2012). Analytical evaluation of the performance of proportional fair scheduling in ofdma-based wireless systems. 2012.

[Aigner and Jüttler, 2009] Aigner, M. and Jüttler, B. (2009). Robust fitting of implicitly defined surfaces using gauss–newton-type techniques. *The Visual Computer*, 25(8):731–741.

[Andradóttir, 1998] Andradóttir, S. (1998). Simulation optimization. In Banks, J., editor, *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*, pages 307–334. Wiley-Interscience.

[Appice et al., 2014] Appice, A., Ciampi, A., Fumarola, F., and Malerba, D. (2014). *Data Mining Techniques in Sensor Networks.* Springer London.

[Aral, 2011] Aral, M. M. (2011). *Environmental Modeling and Health Risk Analysis (Acts/Risk).* Springer-Verlag GmbH.

[Armstrong, 1998] Armstrong, M. (1998). *Basic Linear Geostatistics.* Springer Nature.

[Banks, 1998] Banks, J., editor (1998). *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice.* Wiley-Interscience.

[Barillec et al., 2011] Barillec, R., Ingram, B., Cornford, D., and Csató, L. (2011). Projected sequential gaussian processes: A c++ tool for interpolation of large datasets with heterogeneous noise. *Computers & Geosciences*, 37(3):295–309. Geoinformatics for Environmental Surveillance.

[Barnsley, 2007] Barnsley, M. J. (2007). *Environmental Modeling: A Practical Introduction*. CRC Press.

[Berthouex and Brown, 1994] Berthouex, P. M. and Brown, L. C. (1994). *Statistics for Environmental Engineers*. CRC Press.

[Beven, 2009] Beven, K. (2009). *Environmental modelling: An uncertain future?* CRC Press.

[Birta and Arbez, 2007] Birta, L. G. and Arbez, G. (2007). *Modelling and Simulation: Exploring Dynamic System Behaviour*. Springer.

[Blower et al., 2013] Blower, J. D., Gemmell, A., Griffiths, G. H., Haines, K., Santokhee, A., and Yang, X. (2013). A web map service implementation for the visualization of multidimensional gridded environmental data. *Environmental Modelling & Software*, 47:218–224.

[Box and Draper, 2007] Box, G. E. P. and Draper, N. R. (2007). *Response Surfaces, Mixtures, and Ridge Analyses*. JOHN WILEY & SONS INC.

[Brinkhoff, 2013] Brinkhoff, T. (2013). *Geodatenbanksysteme in Theorie und Praxis*. Wichmann Herbert.

[Brunell, 1992] Brunell, R. M. (1992). An automatic procedure for fitting variograms by cressie's approximate weighted least squares criterion. *Department of Statistical Science Technical Report No. SMU/DS/TR, Southern Methodist University*.

[Burrough et al., 2015] Burrough, P. A., McDonnell, R. A., and Lloyd, C. D. (2015). *Principles of Geographical Information Systems*. Oxford University Press.

[Camara et al., 2014] Camara, G., Egenhofer, M. J., Ferreira, K., Andrade, P., Queiroz, G., Sanchez, A., Jones, J., and Vinhas, L. (2014). Fields as a generic data type for big spatial data. In *International Conference on Geographic Information Science*, pages 159–172. Springer.

[Chiles and Delfiner, 2012] Chiles, J.-P. and Delfiner, P. (2012). *Geostatistics: Modeling Spatial Uncertainty*. John Wiley & Sons, Inc.

[Chun and Griffith, 2013] Chun, Y. and Griffith, D. (2013). *Spatial Statistics and Geostatistics: Theory and Applications for Geographic Information Science and Technology*. SAGE Advances in Geographic Information Science and Technology Series. SAGE Publications.

[Cormen et al., 2005] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2005). *Introduction to Algorithms*. The MIT Press.

[Cornford et al., 2005] Cornford, D., Csató, L., and Opper, M. (2005). Sequential, bayesian geostatistics: a principled method for large data sets. *Geographical Analysis*, 37(2):183–199.

[Couclelis, 1992] Couclelis, H. (1992). People manipulate objects (but cultivate fields): beyond the raster-vector debate in gis. *Theories and methods of spatio-temporal reasoning in geographic space*, pages 65–77.

[Cova and Goodchild, 2002] Cova, T. J. and Goodchild, M. F. (2002). Extending geographical representation to include fields of spatial objects. *International Journal of geographical information science*, 16(6):509–532.

[Craglia et al., 2012] Craglia, M., de Bie, K., Jackson, D., Pesaresi, M., Remetey-Fülöpp, G., Wang, C., Annoni, A., Bian, L., Campbell, F., Ehlers, M., van Genderen, J., Goodchild, M., Guo, H., Lewis, A., Simpson, R., Skidmore, A., and Woodgate, P. (2012). Digital earth 2020: towards the vision for the next decade. *International Journal of Digital Earth*, 5(1):4–21.

[Cressie, 1985] Cressie, N. (1985). Fitting variogram models by weighted least squares. *Journal of the International Association for Mathematical Geology*, 17(5):563–586.

[Cressie, 1990] Cressie, N. (1990). The origins of kriging. *Mathematical Geology*, 22(3):239–252.

[Cressie, 1993] Cressie, N. (1993). *Statistics for Spatial Data*. JOHN WILEY & SONS INC.

[Cressie and Wikle, 2011] Cressie, N. and Wikle, C. K. (2011). *Statistics for Spatio-Temporal Data*. John Wiley and Sons Ltd.

[Dang et al., 2013] Dang, T., Bulusu, N., and Feng, W.-c. (2013). Robust data compression for irregular wireless sensor networks using logical mapping. *ISRN Sensor Networks*, 2013.

[de Smet et al., 2007] de Smet, P. A. M., Horálek, J., and Denby, B. (2007). European air quality mapping through interpolation with application to exposure and impact assessment. In Scharl, A. and Tochtermann, K., editors, *The Geospatial Web: How Geobrowsers, Social Software and the Web 2.0 are Shaping the Network Society (Advanced Information and Knowledge Processing)*. Springer.

[Desassis and Renard, 2013] Desassis, N. and Renard, D. (2013). Automatic variogram modeling by iterative least squares: Univariate and multivariate cases. *Mathematical Geosciences*, 45(4):453–470.

[Duarte and Baraniuk, 2012] Duarte, M. F. and Baraniuk, R. G. (2012). Kronecker compressive sensing. *IEEE Transactions on Image Processing*, 21(2):494–504.

[Ehlers, 2008] Ehlers, M. (2008). Geoinformatics and digital earth initiatives: a german perspective. *International Journal of Digital Earth*, 1(1):17–30.

[Evans, 2003] Evans, E. (2003). *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison-Wesley Professional.

[Ferrari, 1978] Ferrari, D. (1978). *Computer Systems Performance Evaluation*. Prentice Hall.

[Fortier and Michel, 2003] Fortier, P. J. and Michel, H. E. (2003). *Computer Systems Performance Evaluation and Prediction*. Digital Press.

[Gama and Gaber, 2007] Gama, J. and Gaber, M. M., editors (2007). *Learning from Data Streams*. Springer-Verlag Berlin Heidelberg.

[Gama and Pedersen, 2007] Gama, J. and Pedersen, R. U. (2007). Predictive learning in sensor networks. In *Learning from Data Streams*, pages 143–164. Springer.

[Gandibleux et al., 2004] Gandibleux, X., Sevaux, M., Soïensen, K., and T'kindt, V., editors (2004). *Metaheuristics for Multiobjective Optimisation*. Springer Berlin Heidelberg.

[Garnett et al., 2010] Garnett, R., Osborne, M. A., and Roberts, S. J. (2010). Bayesian optimization for sensor set selection. In Abdelzaher, T. F., Voigt, T., and Wolisz, A., editors, *Proceedings of the 9th International Conference on Information Processing in Sensor Networks, IPSN 2010, April 12-16, 2010, Stockholm, Sweden*, pages 209–219. ACM.

[Gelman et al., 2014] Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). *Bayesian data analysis*, volume 2. CRC press Boca Raton, FL.

[Gigch, 1991] Gigch, J. P. V. (1991). *System Design Modeling and Metamodeling*. Springer US.

[Ginevan and Splitstone, 2004] Ginevan, M. E. and Splitstone, D. E. (2004). *Statistical Tools for Environmental Quality Measurement*. Chapman and Hall/CRC.

[Gonzalez and Woods, 2002] Gonzalez, R. C. and Woods, R. E. (2002). *Digital Image Processing (2nd Edition)*. Prentice Hall.

[Goosse, 2015] Goosse, H. (2015). *Climate System Dynamics and Modelling*. Cambridge University Press.

[Gräler et al., 2016] Gräler, B., Pebesma, E., and Heuvelink, G. (2016). Spatio-temporal interpolation using gstat. *R Journal*, 8(1):204–218.

[Gräler et al., 2012] Gräler, B., Rehr, M., Gerharz, L., and Pebesma, E. (2012). Spatio-temporal analysis and interpolation of pm10 measurements in europe for 2009. *ETC/ACM Technical Paper*, 8:1–29.

[Guestrin et al., 2005] Guestrin, C., Krause, A., and Singh, A. P. (2005). Near-optimal sensor placements in gaussian processes. In *Proceedings of the 22nd international conference on Machine learning*, pages 265–272. ACM.

[Guttorp, 2001] Guttorp, P. (2001). Environmental statistics. In Raftery, A. E., Tanner, M. A., and Wells, M. T., editors, *Statistics in the 21st Century*. Chapman and Hall/CRC.

[Havlik et al., 2011] Havlik, D., Schade, S., Sabeur, Z. A., Mazzetti, P., Watson, K., Berre, A. J., and Mon, J. L. (2011). From sensor to observation web with environmental enablers in the future internet. *Sensors*, 11(4):3874–3907.

[Henneböhl et al., 2011] Henneböhl, K., Appel, M., and Pebesma, E. (2011). Spatial interpolation in massively parallel computing environments. In *Proc. of the 14th AGILE International Conference on Geographic Information Science (AGILE 2011)*.

[Huang et al., 2008] Huang, Y., Peng, J., Kuo, C.-C. J., and Gopi, M. (2008). A generic scheme for progressive point cloud coding. *IEEE Transactions on Visualization and Computer Graphics*, 14(2):440–453.

[ín Martínez and Sánchez-Meca, 2010] ín Martínez, F. M. and Sánchez-Meca, J. (2010). Weighting by inverse variance or by sample size in random-effects meta-analysis. *Educational and Psychological Measurement*, 70(1):56–73.

[Isaaks and Srivastava, 1990] Isaaks, E. H. and Srivastava, R. M. (1990). *An Introduction to Applied Geostatistics*. Oxford University Press.

[Jardak et al., 2010] Jardak, C., Riihijärvi, J., Oldewurtel, F., and Mähönen, P. (2010). Parallel processing of data from very large-scale wireless sensor networks. In *Proceedings of the 19th ACM international symposium on high performance distributed computing*, pages 787–794. ACM.

[Jaynes, 2003] Jaynes, E. T. (2003). *Probability theory: The logic of science*. Cambridge university press.

[Jin and Nittel, 2008] Jin, G. and Nittel, S. (2008). Towards spatial window queries over continuous phenomena in sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 19(4):559–571.

[Jorgensen, 1994] Jorgensen, S. (1994). *Fundamentals of Ecological Modelling, Second Edition (Developments in Environmental Modelling).* Elsevier Science.

[Katzfuss and Cressie, 2011] Katzfuss, M. and Cressie, N. (2011). Tutorial on fixed rank kriging (frk) of co2 data. *The Ohio State University: Columbus, OH, USA.*

[Kho et al., 2009] Kho, J., Rogers, A., and Jennings, N. R. (2009). Decentralized control of adaptive sampling in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 5(3).

[Kolo et al., 2012] Kolo, J. G., Shanmugam, S. A., Lim, D. W. G., Ang, L.-M., and Seng, K. P. (2012). An adaptive lossless data compression scheme for wireless sensor networks. *Journal of Sensors*, 2012.

[Kuhn, 2012] Kuhn, W. (2012). Core concepts of spatial information for transdisciplinary research. *International Journal of Geographical Information Science*, 26(12):2267–2276.

[Lantuéjoul, 2002] Lantuéjoul, C. (2002). *Geostatistical Simulation.* Springer Berlin Heidelberg.

[Larman, 2001] Larman, C. (2001). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process (2nd Edition).* Prentice Hall PTR.

[Lavenberg, 1983] Lavenberg, S., editor (1983). *Computer Performance Modeling Handbook (Notes and reports in computer science and applied mathematics).* Academic Press.

[Law, 2014] Law, A. M. (2014). *Simulation Modeling and Analysis.* McGraw-Hill Education - Europe.

[Leinonen et al., 2014] Leinonen, M., Codreanu, M., and Juntti, M. (2014). Compressed acquisition and progressive reconstruction of multi-dimensional correlated data in wireless sensor networks. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6449–6453.

[Li and Heap, 2008] Li, J. and Heap, A. D. (2008). A review of spatial interpolation methods for environmental scientists. Technical report, Geoscience Australia, Australian Government.

[Liang et al., 2016] Liang, Q., Nittel, S., and Hahmann, T. (2016). From data streams to fields: Extending stream data models with field data types. In Miller, J. A., O'Sullivan, D., and Wiegand, N., editors, *Geographic Information Science: 9th International Conference, GIScience 2016, Montreal, QC, Canada, September 27-30, 2016, Proceedings*, pages 178–194. Springer International Publishing.

[Lin et al., 2016] Lin, S., Miao, F., Zhang, J., Zhou, G., Gu, L., He, T., Stankovic, J. A., Son, S., and Pappas, G. J. (2016). Atpc: Adaptive transmission power control for wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 12(1).

[Liu et al., 2012] Liu, B., Zydek, D., Selvaraj, H., and Gewali, L. (2012). Accelerating high performance computing applications: Using cpus, gpus, hybrid cpu/gpu, and fpgas. In *Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2012 13th International Conference on*, pages 337–342. IEEE.

[Lorkowski and Brinkhoff, 2015a] Lorkowski, P. and Brinkhoff, T. (2015a). Environmental monitoring of continuous phenomena by sensor data streams: A system approach based on kriging. In *Proceedings of EnviroInfo and ICT for Sustainability 2015*. Atlantis Press.

[Lorkowski and Brinkhoff, 2015b] Lorkowski, P. and Brinkhoff, T. (2015b). Towards real-time processing of massive spatio-temporally distributed sensor data: A sequential strategy based on kriging. In *Lecture Notes in Geoinformation and Cartography*, pages 145–163. Springer Nature.

[Lorkowski and Brinkhoff, 2016] Lorkowski, P. and Brinkhoff, T. (2016). Compression and progressive retrieval of multi-dimensional sensor data. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B2:27–33.

[Ma, 2007] Ma, C. (2007). Stationary random fields in space and time with rational spectral densities. *IEEE transactions on information theory*, 53(3):1019–1029.

[Matheron, 1988] Matheron, G. (1988). *Estimating and Choosing.* Springer.

[McKillup and Dyar, 2010] McKillup, S. and Dyar, M. D. (2010). *Geostatistics explained: an introductory guide for earth scientists.* Cambridge University Press.

[Medeiros et al., 2014] Medeiros, H. P., Maciel, M. C., Demo Souza, R., and Pellenz, M. E. (2014). Lightweight data compression in wireless sensor networks using huffman coding. *International Journal of Distributed Sensor Networks*, 10(1).

[Mellor and Balcer, 2002] Mellor, S. J. and Balcer, M. J. (2002). *Executable UML: A Foundation for Model-Driven Architecture.* Addison-Wesley Professional.

[Mertins, 1999] Mertins, A. (1999). Signal analysis: Wavelets, filter banks, time-frequency transforms and applications.

[Meyers, 1997] Meyers, J. C. (1997). *Geostatistical Error Management: Quantifying Uncertainty for Environmental Sampling and Mapping (Industrial Engineering).* John Wiley & Sons Inc.

[Michaud, 2001] Michaud, L. M. (2001). Total energy equation method for calculating hurricane intensity. *Meteorology and Atmospheric Physics*, 78(1-2):35–43.

[Müller, 1999] Müller, W. G. (1999). Least-squares fitting from the variogram cloud. *Statistics & Probability Letters*, 43(1):93–98.

[Nagel et al., 2005] Nagel, C., Evjen, B., Glynn, J., Watson, K., Skinner, M., and Jones, A. (2005). *Professional C# 2005.* Wrox.

[Oliver, 1995] Oliver, D. S. (1995). Moving averages for gaussian simulation in two and three dimensions. *Mathematical Geology*, 27(8):939–960.

[Oliver and Webster, 2015] Oliver, M. A. and Webster, R. (2015). *Basic Steps in Geostatistics: the Variogram and Kriging*. Springer-Verlag GmbH.

[Osborne et al., 2012] Osborne, M. A., Roberts, S. J., Rogers, A., and Jennings, N. R. (2012). Real-time information processing of environmental sensor network data using bayesian gaussian processes. *ACM Transactions on Sensor Networks (TOSN)*, 9(1):1.

[Osborne et al., 2008] Osborne, M. A., Roberts, S. J., Rogers, A., Ramchurn, S. D., and Jennings, N. R. (2008). Towards real-time information processing of sensor network data using computationally efficient multi-output gaussian processes. In *Proceedings of the 7th international conference on Information processing in sensor networks*, pages 109–120. IEEE Computer Society.

[Parent and Rivot, 2012] Parent, E. and Rivot, E. (2012). *Introduction to Hierarchical Bayesian Modeling for Ecological Data*. CHAPMAN & HALL.

[Peng et al., 2001] Peng, G., Leslie, L. M., and Shao, Y., editors (2001). *Environmental Modelling and Prediction*. Springer Berlin Heidelberg.

[Pesquer et al., 2011] Pesquer, L., Cortés, A., and Pons, X. (2011). Parallel ordinary kriging interpolation incorporating automatic variogram fitting. *Computers & Geosciences*, 37(4):464 – 473.

[Pollock et al., 1999] Pollock, D. S. G., Green, R. C., and Nguyen, T. (1999). *Handbook of time series analysis, signal processing, and dynamics*. Academic Press.

[Popper, 2002] Popper, K. (2002). *The Logic of Scientific Discovery*. ROUTLEDGE CHAPMAN HALL. Original Edition: 1959.

[Poulton, 2001] Poulton, M., editor (2001). *Computational Neural Networks for Geophysical Data Processing (Handbook of Geophysical Exploration: Seismic Exploration)*. Pergamon.

[Press et al., 2007] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 3 edition.

[Pritsker, 1998] Pritsker, A. A. B. (1998). Principles of simulation modeling. In Banks, J., editor, *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*, pages 31–54. Wiley-Interscience.

[Rasmussen, 2006] Rasmussen (2006). *Gaussian Processes for Machine Learning*. MIT University Press Group Ltd.

[Rigaux et al., 2001] Rigaux, P., Scholl, M., and Voisard, A. (2001). *Spatial Databases*. Elsevier Science & Technology.

[Robert and Casella, 1999] Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer Verlag.

[Rodrigues et al., 2007] Rodrigues, P. P., Gama, J., and Gaber, M. (2007). Clustering techniques in sensor networks. *Learning from Data Streams*, pages 125–142.

[Sabins, 1996] Sabins, F. F. (1996). *Remote Sensing: Principles and Interpretations*. W. H. Freeman.

[Samet, 2006] Samet, H. (2006). *Foundations of Multidimensional and Metric Data Structures*. Elsevier LTD, Oxford.

[Sathe et al., 2013] Sathe, S., Papaioannou, T. G., Jeung, H., and Aberer, K. (2013). A survey of model-based sensor data acquisition and management.

[Schittkowski, 2002] Schittkowski, K. (2002). *Numerical Data Fitting in Dynamical Systems*. Springer.

[Smith, 2007] Smith, C. U. (2007). Introduction to software performance engineering: Origins and outstanding problems. In Bernardo, M. and Hillston, J., editors, *Formal Methods for Performance Evaluation: 7th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, (Lecture Notes in Computer Science)*. Springer.

[Stoica and Moses, 2005] Stoica, P. and Moses, R. L. (2005). *Spectral Analysis of Signals*. Prentice Hall.

[Sun and Sun, 2015] Sun, N.-Z. and Sun, A. (2015). *Model Calibration and Parameter Estimation*. Springer-Verlag.

[Sun et al., 2006] Sun, W., Cetin, M., Thacker, W. C., Chin, T. M., and Willsky, A. S. (2006). Variational approaches on discontinuity localization and field estimation in sea surface temperature and soil moisture. *IEEE Transactions on Geoscience and Remote Sensing*, 44(2):336–350.

[Szyperski, 2002] Szyperski, C. (2002). *Component Software: Beyond Object-Oriented Programming (2nd Edition)*. Addison-Wesley Professional.

[Taylor et al., 2009] Taylor, R. N., Medvidovic, N., and Dashofy, E. (2009). *Software Architecture*. John Wiley and Sons Ltd.

[Tonkin et al., 2016] Tonkin, M. J., Kennel, J., Huber, W., and Lambie, J. M. (2016). Multi-event universal kriging (meuk). *Advances in Water Resources*, 87:92–105.

[Umer et al., 2009] Umer, M., Kulik, L., and Tanin, E. (2009). Spatial interpolation in wireless sensor networks: localized algorithms for variogram modeling and kriging. *GeoInformatica*, 14(1):101–134.

[van den Bos, 2007] van den Bos, A. (2007). *Parameter Estimation for Scientists and Engineers*. JOHN WILEY & SONS INC.

[Vihma et al., 2014] Vihma, T., Pirazzini, R., Fer, I., Renfrew, I. A., Sedlar, J., Tjernström, M., Lüpkes, C., Nygard, T., Notz, D., Weiss, J., et al. (2014). Advances in understanding and parameterization of small-scale physical processes in the marine arctic climate system: a review. *Atmospheric Chemistry and Physics (ACP)*, 14(17):9403–9450.

[Wackernagel, 2003] Wackernagel, H. (2003). *Multivariate Geostatistics*. Springer-Verlag GmbH.

[Wackernagel and Schmitt, 2001] Wackernagel, H. and Schmitt, M. (2001). Statistical interpolation models. In Raftery, A. E., Tanner, M. A., and

Wells, M. T., editors, *Statistics in the 21st Century*, chapter 10. Chapman and Hall/CRC.

[Walkowski, 2010] Walkowski, A. C. (2010). *Modellbasierte Optimierung mobiler Geosensornetzwerke für raumzeitvariante Phänomene.* AKA, Akad. Verlag-Ges.

[Walter, 2011] Walter, M. (2011). *Mathematics for the Environment.* Taylor & Francis Ltd.

[Wang et al., 2006] Wang, W., Pottmann, H., and Liu, Y. (2006). Fitting b-spline curves to point clouds by curvature-based squared distance minimization. *ACM Transactions on Graphics*, 25(2):214–238.

[Webster and Oliver, 2007] Webster, R. and Oliver, M. (2007). *Geostatistics for Environmental Scientists.* Statistics in Practice. Wiley.

[Wei et al., 2015] Wei, H., Du, Y., Liang, F., Zhou, C., Liu, Z., Yi, J., Xu, K., and Wu, D. (2015). A k-d tree-based algorithm to parallelize kriging interpolation of big spatial data. *GIScience & Remote Sensing*, 52(1):40–57.

[Whittier et al., 2013] Whittier, J. C., Nittel, S., Plummer, M. A., and Liang, Q. (2013). Towards window stream queries over continuous phenomena. In *Proceedings of the 4th ACM SIGSPATIAL International Workshop on GeoStreaming*, IWGS '13, pages 2–11, New York, NY, USA. ACM.

[Zeigler et al., 2000] Zeigler, B. P., Praehofer, H., and Kim, T. G. (2000). *Theory of Modeling and Simulation.* Elsevier Science & Technology.

Appendix A

Erklärung über die Eigenständigkeit der erbrachten wissenschaftlichen Leistung

Ich erkläre hiermit, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Andere Personen waren an der inhaltlichen materiellen Erstellung der vorliegenden Arbeit nicht beteiligt.

Insbesondere habe ich hierfür nicht die entgeltliche Hilfe von Vermittlungs- bzw. Beratungsdiensten (Promotionsberater oder andere Personen) in Anspruch genommen. Niemand hat von mir unmittelbar oder mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer anderen Prüfungsbehörde vorgelegt.

....

(Ort, Datum) (Unterschrift)