

# ERROR PROPAGATION IN THE COMPUTATION OF VOLUMES IN 3D CITY MODELS WITH THE MONTE CARLO METHOD

F. Biljecki <sup>a,\*</sup>, H. Ledoux <sup>a</sup>, and J. Stoter <sup>a,b,c</sup>

<sup>a</sup> Delft University of Technology, GIS technology, Delft, The Netherlands - (f.biljecki, j.e.stoter, h.ledoux)@tudelft.nl

<sup>b</sup> Kadaster, Product and Process Innovation, Apeldoorn, The Netherlands

<sup>c</sup> Geonovum, Amersfoort, The Netherlands

## Technical Commission II

**KEY WORDS:** Uncertainty, Error propagation, Monte Carlo Method, 3D city modelling, CityGML, Random3Dcity

### ABSTRACT:

This paper describes the analysis of the propagation of positional uncertainty in 3D city models to the uncertainty in the computation of their volumes. Current work related to error propagation in GIS is limited to 2D data and 2D GIS operations, especially of rasters. In this research we have (1) developed two engines, one that generates random 3D buildings in CityGML in multiple LODs, and one that simulates acquisition errors to the geometry; (2) performed an error propagation analysis on volume computation based on the Monte Carlo method; and (3) worked towards establishing a framework for investigating error propagation in 3D GIS. The results of the experiments show that a comparatively small error in the geometry of a 3D city model may cause significant discrepancies in the computation of its volume. This has consequences for several applications, such as in estimation of energy demand and property taxes. The contribution of this work is twofold: this is the first error propagation analysis in 3D city modelling, and the novel approach and the engines that we have created can be used for analysing most of 3D GIS operations, supporting related research efforts in the future.

## 1 INTRODUCTION

Error propagation in GIS is both an unavoidable fact when processing and utilising datasets, and a field of research. When maps that are stored in a GIS database are used as input to a GIS operation, then the errors in the input will propagate to the output of the operation (Lemmens, 2011). An error in the input dataset propagates differently depending on the used operation, and it is virtually never equal between applications. Therefore the main research question of error propagation is: given the errors in the input to a GIS operation, how large are the errors in its output (Heuvelink, 2005).

Two prominent methods that are used to determine the propagation of errors in GIS operations are the Taylor series method, and the Monte Carlo method. The first method is analytical and it involves establishing mathematical functions that describe the process, while the latter is used with more complex processes, as a numerical brute force method to translate randomly generated uncertainty in inputs to uncertainty in outputs.

This paper investigates how a Monte Carlo simulation can be used to obtain insight into error propagation in 3D city modelling. As we explain in Section 2, generalising this problem from 2D to 3D is more complex than one may expect, which motivated us to research this topic in 3D. In this paper we present an error propagation simulation that we have performed with 3D city models on the operation of volume calculation, which is one of the key operations in 3D GIS, and it is used in a number of use-cases.

Hereby we introduce an engine that creates randomly generated 3D data of buildings. Simulation of input data is not typical for error propagation analyses, however, we motivate our approach by achieving diversity of the input data, and to have better understanding of the complete process. We have implemented a software prototype that creates CityGML models (Open Geospatial Consortium, 2012) in multiple levels of detail (LODs) that are intentionally impaired with normally distributed positional errors

simulated according to 20 accuracy classes (values of standard deviations). For each building, its volume is computed in each LOD and each accuracy class. The erroneous volumes are compared to the ground truth, determined from the error-free synthetic dataset within the corresponding LOD. Afterwards, the root mean square error is computed for each accuracy class and each LOD, along with the relative error in percent. This directly results in determining the relationship between the input and output uncertainty, meeting the main research question of each error propagation analysis.

The results of this work, and the conclusion may serve as a positional accuracy recommendation for use-cases that involve the operation of the volume computation, and for related research efforts in the future. We show that a comparatively small error in the geometry of a 3D city model may cause significant discrepancies in the computation of its volume, and that the relation between input and output uncertainty is linear.

## 2 BACKGROUND AND RELATED WORK

The subject of error propagation is researched in a number of disciplines, such as physics (Taylor, 1997), and has a strong mathematical and experimental foundation that can be applied to other disciplines. It is related to GIS: geographical observations describe phenomena with spatial, temporal, and thematic components that are all acquired with uncertainty (Veregin, 2005). Beside measurement errors, uncertainty may be caused by processing, generalisation and several other factors (Fisher, 2005). Hence, understanding the propagation of errors in GIS is important. For instance, it is important to set expectations when obtaining and utilising datasets for a specific application. One may define a maximum acceptable error of the result of an operation, and by performing the analysis of the propagation of errors within the operation may determine the maximum allowed error in the input data. An example is the usage of 3D city models in estimating the

visibility (line of sight) between two points in space. This technique is used by radio engineers and telecom companies to estimate the radio propagation (Wagen and Rizk, 2003), and for security purposes as in determining the optimal closed-circuit television (CCTV) camera placement (Ying et al., 2002). While the first one may be successfully accomplished with a rough and not overly accurate 3D city model, the latter is focused on street-scale data where a comparatively small error may result in significant errors in the output.

The topic has been researched and documented in a number of publications. Heuvelink et al. (1989) performed a quantitative analysis of gridded 2D data in a raster geographical information systems with two use-cases, Arbia et al. (1998) modelled the error propagation of overlay operations in raster GIS, and Van Oort et al. (2005) researched the propagation of positional uncertainty of vertices in polygons to the computation of its area. Further, Heuvelink and Burrough (1993) investigated error propagation in cartographic modelling using Boolean logic and continuous classification. The documented analyses include the propagation of spatial errors and attribute errors, with the latter being more represented. For instance, Veregin (1995) investigated the propagation of thematic error. Dimension-wise, to the extent of our knowledge there is no work in 3D.

Propagated errors are defined as the discrepancies that exist after performing identical operations on one or more layers of ground truth and the corresponding observed data layers. Error propagation modelling is the formal process of representing the transformations in data quality that occur through GIS operations on data layers (Arbia et al., 1998). The error propagation problem can be formulated mathematically as follows (Heuvelink, 2005). Let  $U(\cdot)$  be the output of a GIS operation  $g(\cdot)$  on the  $m$  input attributes  $A_i(\cdot)$ :

$$U(\cdot) = g(A_1(\cdot), \dots, A_m(\cdot)) \quad (1)$$

The operation  $g(\cdot)$  may represent virtually any GIS operation, for instance, area or slope calculation. The objective of the error propagation analysis is to determine the error in the output  $U(\cdot)$ , given the operation  $g(\cdot)$  and the errors in the input attributes  $A_i(\cdot)$ . An error is therefore a quantitative attribute can be defined as the difference between reality and our representation of reality, and it is usually expressed with the variance of  $U(\cdot)$ .

The two methods to assess the error propagation in GIS are the Taylor series method, and the Monte Carlo method. The rationale of the Taylor method is to approximate the operation with a truncated Taylor series. However, this method is not suited for complex operations, such as the ones used in computational fluid dynamics, since they are difficult to compute analytically.

A Monte Carlo method involves deliberate use of random numbers in a calculation that has the structure of a stochastic process, that is, a sequence of states whose evolution is determined by random events (Kalos and Whitlock, 2008). It is made to work by running the assessment model repeatedly with random disturbances introduced into a selected input factor prior to each run. When used in error propagation, it is used to find the relationship between the distribution of input errors and the model's outputs in order to determine whether input error amplifies or suppresses output error (Emmi and Horton, 1995). Following the notations, the reasoning of the Monte Carlo method is to compute the result of  $g(A_1, \dots, A_m)$  repeatedly, with input values  $A_i$  that are randomly sampled from their joint distribution (Heuvelink, 2005).

Examples of the Monte Carlo method employed in the analysis of error propagation in GIS include an application in flood man-

agement (Qi et al., 2013), a GIS-based assessment of seismic risk (Emmi and Horton, 1995), spatial probabilistic modelling for potential slope failures (Zhou et al., 2003), and an evaluation of the accuracy of agricultural land valuation using land use and the soil information (Fisher, 1991). While error propagation is a mature and studied subject in GIS, state of the art is limited to 2D data. Further, it is focused towards raster data, and research on uncertainty in vector data is seldom. The foundation of the subject is valid also for 3D data and 3D GIS operations, however, such analysis requires further attention. Not only an additional geometric dimension prone to errors is available, but (1) the applications, operations and acquisition methods are different; (2) additional measurements are required and additional geometry is present; and (3) the concept of level of detail becomes significantly more exposed. We elaborate each point in the continuation.

- (1) 3D city models are nowadays used for more than a few dozens applications not possible with 2D data, such as estimation of solar potential of roofs (Hofierka and Zlocha, 2012), and in applications that are considerably different than when used with 2D data, such as estimating noise pollution (Stoter et al., 2008). In contrast with 2D data, they are usually collected with a combinations of acquisition techniques, such as photogrammetry (Hammoudi and Dornaika, 2011), LiDAR (Vosselman and Dijkman, 2001), extrusion from 2D data (Ledoux and Meijers, 2011), and as a generalised conversion from Building Information Models (BIM) (Isikdag and Zlatanova, 2009). As is the case with 2D data, each of the techniques produces spatial data burdened with positional errors, however, a model may be constructed from two techniques that have different accuracy levels, resulting in different positional accuracy for  $x$ ,  $y$  on the one hand, and  $z$  on the other hand.
- (2) The construction of 3D city models generally involves more measurements than the 2D acquisition of the same real-world feature (e.g. height of the eaves of a roof, and position of a window on a wall). Further, 3D city models may contain geometries such as solids and other concepts that are not available in 2D.
- (3) The models may be derived in multiple forms distinguished by levels of detail (LODs), different degrees of the complexity of the spatio-semantic representation that are adapted to multi-scale applications (Stadler et al., 2009; Biljecki et al., 2014b). The LOD concept usually does not cover the spatial accuracy, and when it does impose the requirements, this is frequently overlooked (Biljecki et al., 2013). An arising research question is how error propagation should be analysed for each LOD.

These facts motivated us for this research, and to select the Monte Carlo method to obtain more insight in how errors propagate in a 3D GIS environment. We concentrate on the uncertainties of the geometry of the model, and research how positional accuracy influences the quality of 3D data analysis. Positional accuracy is a prominent data quality element, and a principal descriptor in the metadata of a geo-dataset. In virtually all applications it affects the outcome of a GIS operation, hence we find it relevant for this research.

### 3 METHODOLOGY

In this section we describe the method of this research, along with the implementation in a software prototype, and show examples with data. The methodological process is depicted in Figure 1.

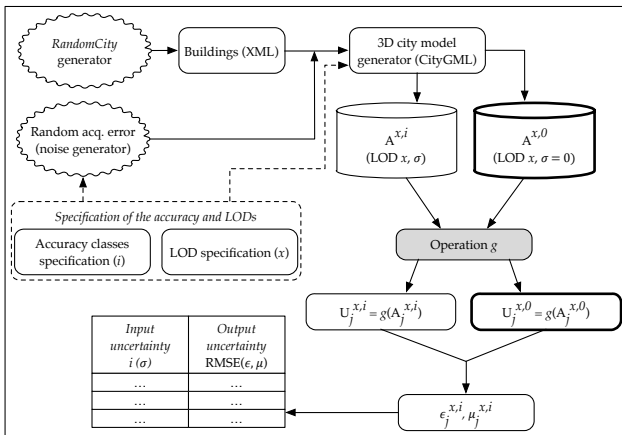


Figure 1: The diagram of the presented methodology. The GIS operation is shown in grey, and the ground truth elements are outlined with thicker strokes. The terms and notations are explained throughout the Section 3.

First we randomly generate properties of synthetic buildings with a novel system that we have designed and implemented (Section 3.1). It is possible to use real-world data, however, the motivation for the development of such a system is to have control over the properties of the buildings in order to generate a high number of dissimilar scenarios in a random way for removing the statistical bias, to avoid geometric construction errors, and to create multi-scale models, which is explained later. This is why we simulated a 3D city model instead of using a real-world data set.

Afterwards, positional noise is added to the geometrical properties of each building, as a simulation of acquisition errors (Section 3.2). As a third step, Section 3.3 describes the realisation of the buildings as 3D city models in the CityGML format in multiple LODs. The process is run iteratively to simulate a number of different accuracy classes (Section 3.4), extending the research for more accuracy ranges.

The fourth step is an operation that uses a 3D city model as input, and outputs a result. We have used the operation of the computation of volumes of buildings.

Note that these steps can be applied to a number of other 3D GIS operations, essentially resulting in this methodology to be considered as a framework with a general application.

The tools used to implement this method are Python, and the Feature Manipulation Engine (FME) of Safe Software Inc.

### 3.1 ENGINE TO GENERATE RANDOM BUILDINGS

In order to understand how positional uncertainty propagates through a 3D GIS operation in a realistic setting, experiments should be done on more than one building. It is essential and a statistical requisite to recreate as many as possible buildings that represent a realistic setting, with different properties. For instance, an equal error at one point in two models of different size, differently affects the operation and causes a different output uncertainty. This is particularly the case in geometry-related operations such as the computation of area and volume.

For this purpose we have designed *Random3Dcity*, an engine that randomises a number of values for a variety of parameters  $p^j$  that define a building  $B_j$ :

$$B_j = \{p_1^j, p_2^j, \dots, p_n^j\} \quad (2)$$

The parameters include (a) attribute ones such roof type, number of windows, and (b) geometrical parameters such as the dimensions of the body of the building, size of windows, their position, etc. We support the four most used types of roofs: gabled, hipped, shed, and flat (Kada, 2007).

The engine mimics the reality by sampling random values from a uniformly distributed probability function. While to the best of our knowledge no research investigated the ranges and distributions of several building properties in the real-world (such as building height or type of roof), for the purpose of this paper we follow the assumption that in general the building properties are uniformly distributed. We have created certain ranges in order to mimic the reality of a residential setting, such as the building width being in between 3 and 10 m. Therefore the values for each building are sampled from the generated probabilistic distribution variable.

With the range of values that we use, it is possible to generate  $3.05 \times 10^{52}$  unique combinations of buildings. This magnitude was determined by calculating the permutations of possible values in the stochastic system.

The values of the parameters were generated as randomly as possible in order to create a set of diverse buildings, however, smaller interventions in the engine in form of constraints have been done. For instance, every building has to have one entrance door, and its height should be at least 2.5 m. These are chosen to make the model as realistic as possible, and to remove unlikely and invalid cases, for instance, building of height of 30 cm, windows bigger than the wall itself, and doors that are 2 m tall but 2 cm wide. This intervention does not affect the result and it rather makes the engine robust conforming to the reality.

After the building properties have been generated, they are written in an XML file containing  $m$  buildings,  $B_1 \dots B_m$ . We exemplify this with one building:

```
<building ID="38b83460-753c-48a8-8c09-52d50131ea0b">
  <origin>173469.0 526427.0 0.0</origin>
  <xSize>4.6</xSize>
  <ySize>4.23</ySize>
  <zSize>3.54</zSize>
  <roof>
    <roofType>Gabled</roofType>
    <h>2.68</h>
    <overhangs>
      <xlength>0.35</xlength>
      <ylength>0.31</ylength>
    </overhangs>
    <dormers>
      <dormer>
        <side>1</side>
        <size>
          <width>1.09</width>
          <height>1.05</height>
        </size>
        <origin>
          <x>1.57</x>
          <y>0.14</y>
        </origin>
      </dormer>
    </dormers>
  </roof>
  <windows>
    <window>
      <wall>1</wall>
      <size>
        <width>1.42</width>
        <height>0.91</height>
      </size>
    </window>
  </windows>
</building>
```

### 3.2 DETERMINATION OF THE ERROR MODEL AND ADDING NOISE TO THE GEOMETRY

The above described dataset is considered as the reality. In order to simulate the error-prone acquisition of the geometry of the

buildings and to create an instance of the dataset that contains errors, we have simulated noise to each parameter  $p_k^j \in B_j$  in an iterative process. In geomatics, each of the dimensions is affected to discrepancies that are normally distributed (Goodchild, 1991; Caspary and Scheuring, 1993). Hence, for each parameter  $p_k^j$ , a degraded parameter  $\hat{p}_k^j$  has been derived by simulating the acquisition error by sampling values from a normal probability distribution function with the standard deviation  $\sigma$ :

$$\hat{p}_k^j = \mathcal{N}(p_k^j, \sigma^2) \quad (3)$$

Hence, we can define  $\epsilon_k^j$ , the simulated error for  $p_k^j$  as the difference between the reality and the (simulated) measurement:

$$\epsilon_k^j = \hat{p}_k^j - p_k^j \quad (4)$$

The parameters  $\hat{p}^j$  comprise the properties of the erroneous version  $\hat{B}_j^\sigma$  of the building  $B_j$ , that are degraded with values sampled from the normal probability distribution function with the standard deviation  $\sigma$ .

Continuing the previous example of the building, new instances of the values of the parameters stored in the XML file have been created. The below excerpt of the XML are the erroneous parameters with the simulated noise of  $\sigma = 0.1$  m for each parameter.

```
<xSize>4.69</xSize>
<ySize>4.19</ySize>
<zSize>3.42</zSize>
```

The motivation for deriving the geometric properties mostly in the form of distances (e.g. height of the roof), rather than positions of points (e.g. coordinates of the southern tip of the roof) is practical, and here the benefits of this approach are exposed. Most software packages nowadays facilitate modelling of buildings by measuring the distances of edges at right angles (Loch-Dehbi and Plümer, 2011), especially in automatic workflows (Fischer et al., 1998). As the noise is added to the lengths of the edges, the right angles in the geometry are preserved. This is in contrast to separately adding noise at vertices as it would break right angles, it could potentially disrupt geometric constructions such as solids, and make the consequent process unnecessary more complex.

### 3.3 CITYGML MODEL REALISATION

In this step we generate the 3D city model of each building  $B_j$  and its erroneous instance  $\hat{B}_j^\sigma$  based on the properties  $p^j$  and  $\hat{p}^j$ . We have decided to generate the models in the international standard CityGML published by the OpenGeospatial Consortium (OGC) because it is one of the relevant standards in 3D city modelling, gradually being adopted by the community, and it enables the storage of models comprising complex semantics along with a structured geometry (Gröger and Plümer, 2012; Stadler and Kolbe, 2007). The standard defines five LODs that roughly reflect the model's complexity. Buildings may be represented in LOD0 by footprint and/or roof edge polygons. LOD1 is a block model comprising prismatic buildings with flat roof structures. In contrast, a building in LOD2 has differentiated roof structures. LOD3 denotes architectural models with detailed wall and roof structures potentially including doors and windows. LOD4 completes a LOD3 model by adding interior structures for buildings (Kolbe et al., 2009; Open Geospatial Consortium, 2012).

Because of the GIS operation that we use in this paper (volume computation), we find LOD0 and LOD4 not applicable, so we focus on the LODs 1, 2 and 3.

We have implemented a software prototype that generates four LOD CityGML representations for each building based on the parameters presented in Section 3.1. The four representations are as follows: (1) LOD1 solid, a block model, where the height of the top represents the half of the height of the roof. This is in line with current modelling practices (Ordnance Survey, 2014); (2) LOD2 solid, a model with a basic roof shape; (3) LOD3 solid, an enhanced model including superstructures such as dormers. Chimneys, antennas, and other building elements that do not contribute towards the volume of the building are not used in the generation of the solid; and (4) another LOD3, as a semantic "full model", which is represented by a set of surfaces. This representation is not essential since the volume computation is not performed on it, but it is generated for a more detailed inspection of the buildings generated in the stochastic process, and for operations left for future work.

The four generated datasets of a building are shown in Figure 2 as an example. This example is in harmony with the previous examples: it shows the generated models of the building whose XML specifications are given in the Section 3.1.

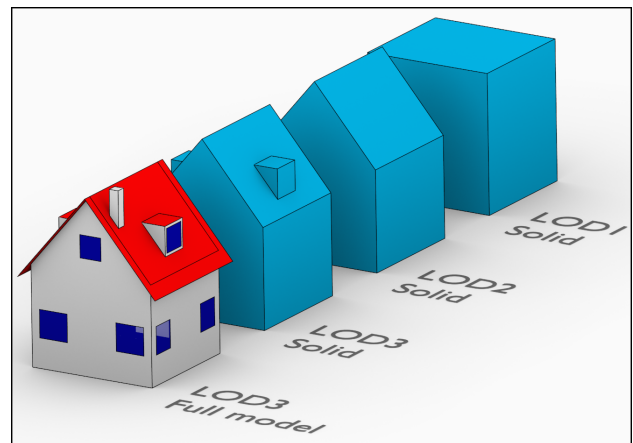


Figure 2: The example of the realisation of multiple LODs in CityGML of one building, from the parameters randomly generated by *Random3Dcity*. This is the building in the XML file that is shown as the example through Sections 3.1 and 3.2.

For notational purposes, the CityGML representation of the building  $\hat{B}_j^\sigma$  in LOD  $x$  is denoted with  $A_j^{x,\sigma}$ .

For this research, a dataset containing 40 000 buildings has been generated, and its fragment is shown in Figure 3, with a close-up in Figure 4.

### 3.4 MULTIPLE ACCURACY CLASSES

Section 3.2 described the simulation of errors by sampling a normally distributed probability function with a value of  $\sigma$ .

In error propagation, one of the relevant research questions is how much different grades of accuracy affect the output of an operation (cf. Section 2). This also helps in determining the required accuracy of the acquisition of the data in order to achieve acceptable results in a GIS workflow.

Therefore, we have considered multiple accuracy classes, and generated noise according to the properties of each. We have defined 20 accuracy classes  $i$  with varying standard deviations

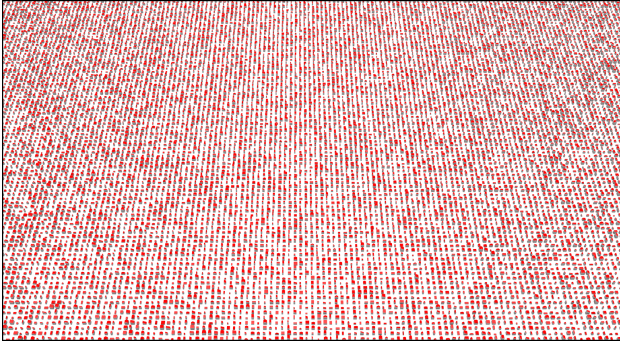


Figure 3: The CityGML realisation of the XML dataset of the randomly generated buildings. The dataset contains 40 000 buildings (in a grid  $200 \times 200$ ).

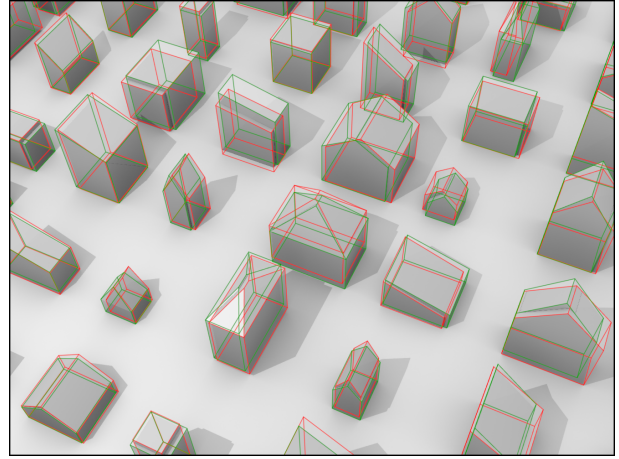


Figure 5: Visualisation of the LOD2 of the ground truth (green), and the LOD2 impaired by normally distributed positional errors with  $\sigma = 0.4$  m (red model). Equal spatial extent as in Fig. 4.



Figure 4: Visualisation of a small spatial extent of the LOD3 dataset that is a semantically enriched boundary representation. The colours represent the different semantic classes.

$\sigma_i$ , ranging from  $\sigma_1 = 0.05$  m to  $\sigma_{20} = 1.00$  m in increments of 0.05 m. In an iterative process, for each accuracy class, each parameter  $p_k^j$  is added a simulated acquisition error, resulting in 20 erroneous parameters  $\hat{p}_k^{j,i}$ .

After the simulation of noise, the models were realised with the CityGML generator yielding four datasets for each accuracy class (LOD1<sup>i</sup>, LOD2<sup>i</sup>, and two instances of LOD3<sup>i</sup>), and additional four for the error-free (ground truth,  $i = 0$ ) specifications (LOD1<sup>0</sup>, LOD2<sup>0</sup>, and two LOD3<sup>0</sup>). Therefore, each building version  $B_j^i$  is represented with four models  $A_j^{x,i}$ , where  $x$  is the LOD.

Due to finer detail, for LOD3 the accuracy classes are limited to 0.3 m. This results in 56 datasets (2.2M representations of 40k buildings). An example of two LOD2 datasets, the ground truth and one where  $i = 8$  ( $\sigma = 0.4$  m) are shown overlapped in Fig. 5.

### 3.5 COMPUTING AND COMPARING VOLUMES

The important part of every error propagation analysis is the considered GIS operation  $g$  or set of operations  $g_1, g_2, \dots, g_n$ . For a proper error propagation analysis, the pre-requisite of an operation is that its results are measurable quantitative attributes, i.e. that it yields values that are on a rational scale. For instance, the outcome of using 3D city models in applications such as navigation and urbanism can hardly be quantified in such a way. On the other hand, estimating solar potential of a rooftop yields a quantitative value, i.e. the annual global solar radiation (kWh/y) (Catita et al., 2014).

We discuss the operation of volume computation of buildings, which is essential in use-cases such as energy demand estimation (Perez et al., 2013), determination of property taxes (Boeters, 2013), and volumetric visibility analysis of urban environments (Fisher-Gewirtzman et al., 2013). Our motivation for the selection of this operation, is that it is one of the fundamental 3D operations, and it is practically simple to implement. Further, from the implementation perspective we are also restricted by the limited number of operations that enable us to automatically load, compute and analyse a vast amount of data, which is not the case in the computation of volumes. From now on, we annotate this operation with  $g$ , which as an input takes a model  $A_j$ , and performs the volume computation generating an output  $U_j$ . Since this is made for multiple LODs  $x$  and in a series of accuracy classes  $i$ , the full relation is:

$$U_j^{x,i} = g(A_j^{x,i}) \quad (5)$$

The results of the computations from FME automated with Python are stored in XML files, which are consequently automatically analysed by our software prototype.

After the volume has been computed for each representation, the error  $\epsilon_j^{x,i}$  is calculated as the difference between the computed volume and the ground truth (model where  $i = 0$ ), which is the volume of the model in the LOD  $x$  and  $\sigma = 0$ :

$$\epsilon_j^{x,i} = \hat{U}_j^{x,i} - U_j^{x,0} \quad (6)$$

Further, for each representation, the relative error  $\mu_j^{x,i}$ , which is the division of the volumetric error with the ground truth volume, has been calculated:

$$\mu_j^{x,i} = \frac{\epsilon_j^{x,i}}{U_j^{x,0}} \quad (7)$$

This measure may be expressed in percent. Considering the relative error is important in order to remove the bias from the volume of the building, as an equal error in the input generally causes a different error in the output depending on the properties (i.e. size) of the model.

Finally, in each LOD  $x$  and in each accuracy class  $i$ , the root mean square error (RMSE) of the operation was derived in order to assess the propagated uncertainty of the operation:

$$\text{RMSE}(\epsilon^{x,i}) = \sqrt{\frac{\sum_{j=1}^m (\epsilon_j^{x,i})^2}{m}} \quad (8)$$

This is also done for the relative dimension-less errors:

$$\text{RMSE}(\mu^{x,i}) = \sqrt{\frac{\sum_{j=1}^m (\mu_j^{x,i})^2}{m}} \quad (9)$$

The association between the input uncertainty  $i$  ( $\sigma_i$ ) and the resulting RMSE values (absolute and relative) is the main product of this error propagation analysis. It enables us to determine how much the input error affects the error in the output of the operation of the building volume computation, and to find their relationship.

#### 4 RESULTS OF THE ERROR PROPAGATION ANALYSIS

In this section we present and interpret the results of the method on a sample of 40000 buildings, and derive conclusions. As a form of validation, we have run the software prototype multiple times, with multiple values of  $m$ , and we have obtained comparable results with negligible deviations. The value  $m = 40000$  proved high enough to produce minimal deviations when ran multiple times. The building randomiser generated diverse buildings, with their volume ranging from 28.62 to 1185.75 m<sup>3</sup>, ensuring a wide range of values, as is the case in reality.

The main outcome of this analysis, the relationship between the input and the output uncertainty in the computations of building volumes, is shown in the Tables 1 and 2. The first table shows the RMSE of the error  $\epsilon^{x,i}$ , and it is expressed in cubic metres. The latter shows the RMSE of the relative output errors  $\mu^{x,i}$ , as defined in Section 3.5, and it is expressed in percent. Due to limited space, we give the data in the tables for 10 accuracy classes, without odd  $i$  values (in  $\sigma = 0.1$  m increments). This is also shown graphically in Figure 6, however, the graph contains all the accuracy classes. From the results we conclude that the relation is linear in both of the RMSE values, i.e. that the increase of the positional uncertainty in the input propagates linearly to the uncertainty in the output. Depending on the intended use of the operation, this relation may be found as significant. It is shown that a comparatively small error in the input data may cause a relatively significant error in the output.

The second relevant finding is the distribution of errors  $\epsilon^{x,i}$  within the same LOD and accuracy class. Figures 7-9 show the distribution of output errors for each LOD separately, and for two accuracy classes. First of all, the output errors are not normally distributed, as one might have anticipated from the errors in the input data which are sampled from the normal probability distribution function. This was concluded by fitting a normal distribution function based on the computed values, which was not achieved (Fig. 8). It was also disproved by a normality test developed by D'Agostino (1971) that combines skew and kurtosis to produce an omnibus test of normality. After tests, we have realised that the Student's t-distribution fits the distribution of the errors, hence we can conclude that the output errors are distributed according to it. An example of the distribution of relative errors  $\mu^{x,i}$  is not included because it is congruent with these examples.

Input uncertainty		Output uncertainty (RMSE [m <sup>3</sup> ])		
Class $i$	$\sigma$ [m]	LOD1	LOD2	LOD3
0	0.0	0.00	0.00	0.00
2	0.1	9.23	9.18	9.15
4	0.2	18.18	18.22	18.31
6	0.3	27.52	27.39	27.24
8	0.4	36.32	36.46	
10	0.5	45.83	45.67	
12	0.6	55.13	55.04	
14	0.7	64.26	64.10	
16	0.8	74.15	72.46	
18	0.9	82.18	82.72	
20	1.0	91.98	91.83	

Table 1: The RMSE( $\epsilon^{x,i}$ ) for each accuracy class and LOD. The values are computed also for the ground truth models ( $i = 0$ ) for self-validation purposes. The simulations in LOD3 were performed up to and including  $\sigma = 0.3$  m due to the fine LOD.

Input uncertainty		Output uncertainty (RMSE [%])		
Class $i$	$\sigma$ [m]	LOD1	LOD2	LOD3
0	0.0	0.00	0.00	0.00
2	0.1	3.10	3.13	3.14
4	0.2	6.19	6.25	6.23
6	0.3	9.33	9.14	9.40
8	0.4	12.36	12.48	
10	0.5	15.54	15.56	
12	0.6	18.75	18.85	
14	0.7	21.87	21.93	
16	0.8	24.96	25.02	
18	0.9	28.15	28.32	
20	1.0	31.40	31.70	

Table 2: The RMSE( $\mu^{x,i}$ ), expressed in percent, for each accuracy class and LOD.

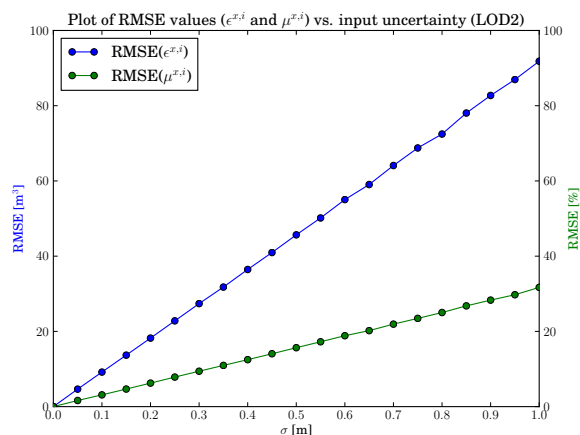


Figure 6: Root mean squared error values of the volume for LOD2 ( $m = 40000$ ). The plot shows that the relation between the input and output uncertainty is linear in both RMSE values.

The non-normal distribution could be explained by the distribution of the ground truth volumes. We have realised that they are not uniformly distributed, despite the uniform distribution of the geometric parameters that the buildings are comprised of. We plan to investigate this in future work by expanding the analysis by creating volume brackets, i.e. running the analysis on classes with different buildings of an approximately equal volume.

The third important result is the independence between the output

uncertainty and the LOD. The RMSE values for each are, neglecting smaller deviations, equal, which makes the error propagation in this operation independent of the LOD.

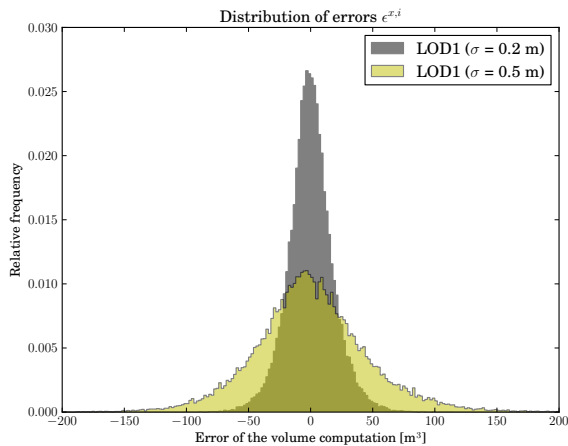


Figure 7: Distribution of volume errors in the accuracy classes with  $\sigma = 0.2$  and  $0.7$  for LOD1.

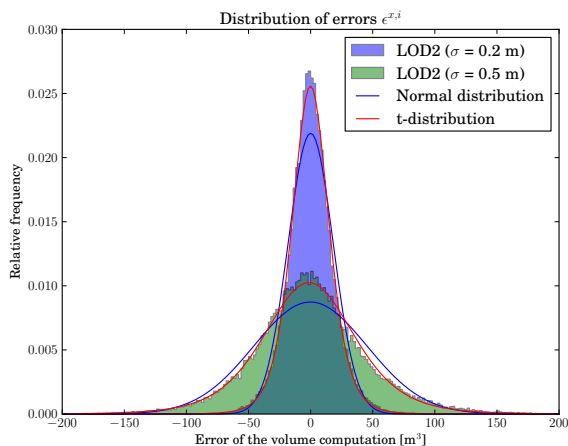


Figure 8: Distribution of volume errors in the accuracy classes  $\sigma = 0.2$  and  $0.7$  for LOD2, with the attempts of fitting two probability distribution functions.

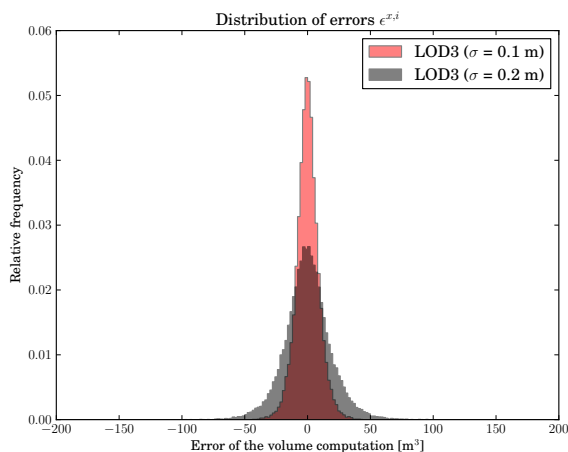


Figure 9: Distribution of volume errors in the accuracy classes  $\sigma = 0.1$  and  $0.2$  for LOD3. Note that here the selection of the classes is different, since in LOD3 they have been limited.

## 5 CONCLUSIONS AND FUTURE WORK

In this research we have performed an error propagation analysis involving 3D city models and volume computation of buildings. The input values are drawn from the assumed uniformly probability distributions of the geometric and other properties of the buildings. The important findings are the distribution of the resulting volume errors, and that the root mean square error linearly increases as the uncertainty in the input increases. We have computed the RMSE for each of the 20 simulated accuracy classes and 3 LODs, establishing the relation between the input and output uncertainty. Further, we have discovered that in this operation each LOD is equally affected by positional errors.

The results of this work may be used by practitioners that rely on 3D city models for computing the volumes of buildings. For instance, the procedure for determining the property taxes based on the volume of the building may have a regulation that the highest allowed uncertainty of the calculated volume is 3%. By using our work, practitioners could determine that the allowed uncertainty of the positions in the input data should be no more than 0.1 m.

The principal contributions are the design and implementation of a robust building randomiser, its CityGML model realisation, and an error propagation analysis in 3D GIS performed with the Monte Carlo method on vector data. The framework and the engine that we have created can be used for future work in this field for a number of other use-cases, and it is not limited to positional uncertainty—it can include also attributes and temporal aspects.

Since multi-LOD datasets are seldom in practice (Biljecki et al., 2014a), additional strengths of using the developed engine are the unaffectedness from such shortcoming, and the availability of these datasets for purposes beyond error propagation analyses.

For future work we plan to extend this research by involving multiple chained operations forming complete use-cases, and to create distinct accuracy classes depending on the features. For instance, roof and wall features are usually acquired with different acquisition techniques that have different accuracy capabilities.

## ACKNOWLEDGMENTS

We thank Safe Software Inc. for providing us with a licence for FME. This research is supported by the Dutch Technology Foundation STW, which is part of the Netherlands Organisation for Scientific Research (NWO), and which is partly funded by the Ministry of Economic Affairs. (Project code: 11300)

## References

- Arbia, G., Griffith, D. and Haining, R., 1998. Error propagation modelling in raster GIS: overlay operations. *International Journal of Geographical Information Science* 12(2), pp. 145–167.
- Biljecki, F., Ledoux, H. and Stoter, J., 2014a. Improving the consistency of multi-LOD CityGML datasets by removing redundancy. In: *Lecture Notes in Geoinformation and Cartography. Proceedings of the 9th 3D GeoInfo Conference, Dubai, UAE.*
- Biljecki, F., Ledoux, H., Stoter, J. and Zhao, J., 2014b. Formalisation of the level of detail in 3D city modelling. *Computers, Environment and Urban Systems* 48, pp. 1–15.
- Biljecki, F., Zhao, J., Stoter, J. and Ledoux, H., 2013. Revisiting the concept of level of detail in 3D city modelling. In: *Proceedings of the ISPRS 8th 3D GeoInfo Conference & WG II/2 Workshop, Istanbul, Turkey, pp. 63–74.*

- Boeters, R., 2013. Automatic enhancement of CityGML LoD2 models with interiors and its usability for net internal area determination. Master's thesis, Delft University of Technology.
- Casparly, W. and Scheuring, R., 1993. Positional accuracy in spatial databases. *Computers, Environment and Urban Systems* 17(2), pp. 103–110.
- Catita, C., Redweik, P., Pereira, J. and Brito, M. C., 2014. Extending solar potential analysis in buildings to vertical facades. *Computers and Geosciences* 66, pp. 1–12.
- D'Agostino, R. B., 1971. An omnibus test of normality for moderate and large size samples. *Biometrika* 58(2), pp. 341–348.
- Emmi, P. C. and Horton, C. A., 1995. A Monte Carlo simulation of error propagation in a GIS-based assessment of seismic risk. *International Journal of Geographical Information Systems* 9(4), pp. 447–461.
- Fischer, A., Kolbe, T. H., Lang, F., Cremers, A. B., Förstner, W., Plümer, L. and Steinhage, V., 1998. Extracting Buildings from Aerial Images Using Hierarchical Aggregation in 2D and 3D. *Computer Vis. and Image Understanding* 72(2), pp. 185–203.
- Fisher-Gewirtzman, D., Shashkov, A. and Doytsher, Y., 2013. Voxel based volumetric visibility analysis of urban environments. *Survey Review* 45(333), pp. 451–461.
- Fisher, P. F., 1991. Modelling soil map-unit inclusions by Monte Carlo simulation. *International Journal of Geographical Information Systems* 5(2), pp. 193–208.
- Fisher, P. F., 2005. Models of uncertainty in spatial data. In: *Geographical Information Systems. Principles, Techniques, Management and Applications*, Wiley, pp. 191–205.
- Goodchild, M. F., 1991. Issues of quality and uncertainty. In: *Advances in cartography*, Elsevier, pp. 113–139.
- Gröger, G. and Plümer, L., 2012. CityGML – Interoperable semantic 3D city models. *ISPRS Journal of Photogrammetry and Remote Sensing* 71, pp. 12–33.
- Hammoudi, K. and Dornaika, F., 2011. A Featureless Approach to 3D Polyhedral Building Modeling from Aerial Images. *Sensors* 11(1), pp. 228–259.
- Heuvelink, G., 2005. Propagation of error in spatial modelling with GIS. In: *Geographical Information Systems. Principles, Techniques, Management and Applications*, Wiley, pp. 207–217.
- Heuvelink, G. B., Burrough, P. A. and Stein, A., 1989. Propagation of errors in spatial modelling with GIS. *International Journal of Geographical Information Science* 3(4), pp. 303–322.
- Heuvelink, G. B. M. and Burrough, P. A., 1993. Error propagation in cartographic modelling using Boolean logic and continuous classification. *International Journal of Geographical Information Systems* 7(3), pp. 231–246.
- Hofierka, J. and Zlocha, M., 2012. A New 3-D Solar Radiation Model for 3-D City Models. *Transactions in GIS* 16(5), pp. 681–690.
- Isikdag, U. and Zlatanova, S., 2009. Towards Defining a Framework for Automatic Generation of Buildings in CityGML Using Building Information Models. In: *3D Geo-Information Sciences*, Springer, pp. 79–96.
- Kada, M., 2007. Scale-dependent simplification of 3D building models based on cell decomposition and primitive instancing. In: *Proceedings of the 8th International Conference on Spatial Information Theory*, Melbourne, Australia, pp. 222–237.
- Kalos, M. H. and Whitlock, P. A., 2008. *Monte Carlo Methods*. 2 edn, Wiley.
- Kolbe, T. H., Nagel, C. and Stadler, A., 2009. CityGML–OGC Standard for Photogrammetry. In: D. Fritsch (ed.), *Proceedings of the 52nd Photogrammetric Week '09*, Stuttgart, Germany, pp. 265–277.
- Ledoux, H. and Meijers, M., 2011. Topologically consistent 3D city models obtained by extrusion. *International Journal of Geographical Information Science* 25(4), pp. 557–574.
- Lemmens, M., 2011. Quality of Geo-information. In: *Geo-information. Technologies, Applications and the Environment*, Springer Netherlands, Dordrecht, pp. 211–227.
- Loch-Dehbi, S. and Plümer, L., 2011. Automatic reasoning for geometric constraints in 3D city models with uncertain observations. *ISPRS Journal of Photogrammetry and Remote Sensing* 66(2), pp. 177–187.
- Open Geospatial Consortium, 2012. OGC City Geography Markup Language (CityGML) Encoding Standard 2.0.
- Ordnance Survey, 2014. New building height data released. Press release published on 18 Mar 2014.
- Perez, D., Kämpf, J. H. and Scartezzini, J.-L., 2013. Urban Area Energy Flow Microsimulation for Planning Support: a Calibration and Verification Study. *International Journal On Advances in Systems and Measurements* 6(3&4), pp. 260–271.
- Qi, H., Qi, P. and Altinakar, M. S., 2013. GIS-Based Spatial Monte Carlo Analysis for Integrated Flood Management with Two Dimensional Flood Simulation. *Water Resources Management* 27(10), pp. 3631–3645.
- Stadler, A. and Kolbe, T. H., 2007. Spatio-semantic coherence in the integration of 3D city models. In: *Proceedings of the WG II/7 5th International Symposium Spatial Data Quality*, Enschede, the Netherlands, p. 8.
- Stadler, A., Nagel, C., König, G. and Kolbe, T. H., 2009. Making Interoperability Persistent: A 3D Geo Database Based on CityGML. In: *3D Geo-Information Sciences*, Springer, pp. 175–192.
- Stoter, J., de Kluijver, H. and Kurakula, V., 2008. 3D noise mapping in urban areas. *International Journal of Geographical Information Science* 22(8), pp. 907–924.
- Taylor, J. R., 1997. *An Introduction to Error Analysis. The Study of Uncertainties in Physical Measurements*, 2 edn, University Science Books.
- van Oort, P., Stein, A., Bregt, A. K., de Bruin, S. and Kuipers, J., 2005. A variance and covariance equation for area estimates with a geographic information system. *Forest Science* 51(4), pp. 347–356.
- Veregin, H., 1995. Developing and testing of an error propagation model for GIS overlay operations. *International Journal of Geographical Information Systems* 9(6), pp. 595–619.
- Veregin, H., 2005. Data quality parameters. In: *Geographical Information Systems. Principles, Techniques, Management and Applications*, John Wiley & Sons, pp. 177–189.
- Vosselman, G. and Dijkman, S., 2001. 3D building model reconstruction from point clouds and ground plans. In: *Proceedings of the workshop on Land surface mapping and characterization using laser altimetry*, Annapolis, Maryland, USA, pp. 37–44.



Wagen, J.-F. and Rizk, K., 2003. Radiowave propagation, building databases, and GIS: anything in common? A radio engineer's viewpoint. *Environment and Planning B: Planning and Design* 30(5), pp. 767–787.

Ying, M., Jingjue, J. and Fulin, B., 2002. 3D-City Model supporting for CCTV monitoring system. In: *Proceedings of the IS-*

*PRS Commission IV Symposium on Geospatial Theory, Processing and Applications*, Ottawa, Canada, p. 4.

Zhou, G., Esaki, T., Mitani, Y., Xie, M. and Mori, J., 2003. Spatial probabilistic modeling of slope failure using an integrated GIS Monte Carlo simulation approach. *Engineering Geology* 68(3-4), pp. 373–386.