ORIGINAL PAPER

CrossMark

A semi-automatic method for 3D modeling and visualizing complex geological bodies

Renbo Song [1] · Xiaoqian Qin [1] · Yeqing Tao [1] · Xiyuan Wang [1] · Biao Yin [1] · Yuexiang Wang [1] · Wenhui Li [1]

Abstract

The technology of 3D geological modeling is a powerful tool for representing the geological structures and subsurface strata distribution, which can effectively assist in the geological interpretation and analysis work. However, most existing modeling methods usually need the supports of professional geological modeling software. These systems often have tedious data preprocessing, complex modeling and visualization processes, and non-professionals have difficulty mastering them in a short time. In order to improve the operability and intuitiveness for 3D geological modeling and visualization, this paper proposes a semi-automatic method for 3D modeling and visualizing complex geological bodies by combining typical GIS systems and 3D modeling software such as ArcGIS and SketchUp. First, the Python object-oriented program language and Arcpy package are combined for preprocessing borehole data automatically. Next, the multiple 3D analysis and file conversion tools of Arctoolbox and ModelBuilder are combined for constructing a model to generate strata interfaces. Then, the makefaces and Boolean plug-ins of SketchUp are combined for semi-automatically constructing the 3D model of complex geological bodies and implementing various forms of visualization. Last, the borehole data and geological maps acquired from the geological survey are used for testing our modeling method. The modeling processes and results show that the method can quickly construct the 3D model of complex geological bodies with very complex geological structures such as missing strata, faults and lenses. This paper shows the simplicity and practicality of the proposed method for modeling and visualizing complex geological bodies.

Keywords 3D geological modeling · Complex geological bodies · GIS · Visualization

Introduction

In recent years, the technology of three-dimensional (3D) geological modeling has attracted much more attention from the field of mining, geology, surveying and mapping, geographic information, engineering construction and other geoscience fields since it can more intuitively help interpret the 3D geological structures and spatial distributions of underground rock than the technology of traditional two-dimensional (2D) geological mapping (Alan and Norman 2003; Martin-Izard et al. 2015; Kaufmann and Martin 2008; Xiao et al. 2015; Choi et al. 2009; Gong et al. 2004; Galera et al. 2003; Hou et al. 2016). However, most existing modeling methods

✉ Renbo Song
  srb_id2000@aliyun.com

[1] School of Urban and Environmental Science of Huaiyin Normal University, Huaian 223300, China

usually need the supports of a professional geological modeling system such as GOCAD (Zanchi et al. 2009) and FLAC3D (Xu et al. 2011). The associated data preprocessing is very tedious and their operation processes are very complex, and non-professional users must spend much time training to master them. Meanwhile, the prices of commercial geological modeling systems are such that high- and small-scale enterprises can not afford to buy them. Some researchers and experts seek to develop 3D geological modeling software by 3D graphic platforms such as OpenGL (Wang and Huang 2012; Wu et al. 2005; Zhu et al. 2012) and Java3D (Yang et al. 2006). This kind of modeling method is often very time-consuming and costly, and users not only have to design complex data structures and algorithms from scratch, but also have to spend much time on mastering many programming skills. Meanwhile, the stability and availability of the constructed systems are often very poor compared with commercial 3D geological modeling software systems. In addition, some experts have constructed simple 3D geological data visualization application (Mccarthy and Graniero 2006; Choi et al. 2009)

and even developed 3D modeling systems based on the existing GIS system such as ArcGIS and MapGIS (Zhu et al. 2006). The defects of existed modeling methods have extremely limited its further promotion in the fields of geosciences research and engineering. Especially, complex geological objects are often involved in the fields of mining, geology and engineering construction, it is very necessary to develop a simple and practical modeling method for meeting the wide requirements of the above fields.

At present, great progress has been made based on the foundational theories (Houlding 1994; Mallet 2002), but it is still challenging work to achieve fully automatic 3D geological modeling and visualization since there remain many difficulties when dealing with various geological formations within a single software system. Through systematically analyzing the existing modeling methods and their implementations, their operation processes mainly include three phases: data preprocessing, modeling and visualization. Meanwhile, many technicians have been accustomed to managing their geological data by using the traditional GIS systems. Meanwhile, they prefer creating 3D models with 3D modeling software such as AutoCAD (Tonini et al. 2008) and 3dsMax (Wang et al. 2009). In fact, the whole processes of data processing, modeling and visualization can be performed by combining the above GIS systems and 3D modeling software, which can not only easily manage and analyze the geological data, but also guarantee the efficiency and reliability of the modeling results.

Furthermore, considering comprehensively from the perspective of the software market occupancy, functionality and usability of existing GIS systems, ArcGIS is the most widely used GIS system. ArcGIS not only has the basic functions for data processing and mapping, but also provides the powerful capabilities for modeling, visualization and spatial analysis. Especially, its embedded Arcpy site package can be combined with the Python object-oriented programming language and the visual modeling tools of ModelBuilder, which can greatly improve the efficiency of data processing and modeling. Meanwhile, SketchUp is simple and practical 3D modeling software directly oriented to the architectural scheme designs, and it is so easy to use that it can be mastered in a very short time. In addition, it provides a lot of auxiliary drawing plug-ins, which can greatly improve the modeling efficiency and make up for the shortages of ArcGIS for modeling complex geological objects.

In summary, in order to promote the further popularization in the fields of mining, geology and engineering construction widely, this paper attempts to explore a semi-automatic method for 3D modeling and visualizing complex geological bodies by combing a typical GIS system and 3D modeling software such as ArcGIS and SketchUp. The main merits are brought out to make the modeling and visualization processes more convenient, intuitive and reliable, and the main

implementing technologies related to data preprocessing, modeling and visualization are discussed in detail.

Methodology

The term geological body is a loose term commonly used in geological fields, usually referring to earth crusts occupying a certain space, with inherent ingredients that can be distinguished from the surrounding geological materials. The shape and spatial distribution of geological bodies are often very complex because they have gone through extensive periods of geological action and processes, but they usually can be classified into stratified and non-stratified geological bodies (Li 2000). The general model for describing geological bodies can be classified into four types (Wu et al. 2002), which mainly include wireframe-based models (Xu and Tian 2009), surface-based models (Lázaro et al. 2014), volume-based models (Wu 2004; Gong et al. 2004) and surface-volume mixed models (Cheng et al. 2005).

Complex geological bodies can be seen as a series of closed geometrical strata objects divided by different structural surfaces such as strata interfaces, fault surfaces and modeling boundary surfaces, and can be described with the boundary representation (B-rep) model. The main modeling ideas of this paper reference the modeling method of constructive solid geometry (CSG) for building complex geological bodies, and its principle and implementation can be described as follows: First, complex geological bodies are decomposed into a series of simple strata objects such as strata interfaces (Fig. 1a), fault surfaces and modeling boundary surfaces (Fig. 1b); Next, a CSG tree is constructed for describing the complex geological bodies with a series of simple strata bodies (Fig. 1c) and faults bodies (Fig. 1d). Then, some Boolean operations such as Boolean (∪), intersection (∩) and difference (−) are combined for generating a 3D model of complex geological bodies (Fig. 1e). Last, the above operations can be easily implemented by the Boolean plug-ins of SketchUp; its modeling principle and implementing method are shown in Fig. 1.

Implementation of modeling and visualization

Designing the work flows

The work flows designed for data preprocessing, modeling and visualization are based on the existing modeling methods, but are different from them since the operability, intuitiveness and practicability are considered adequately. Meanwhile, the special situations of missing strata and local lenses are dealt with at the stage of data processing, and the fault formation is
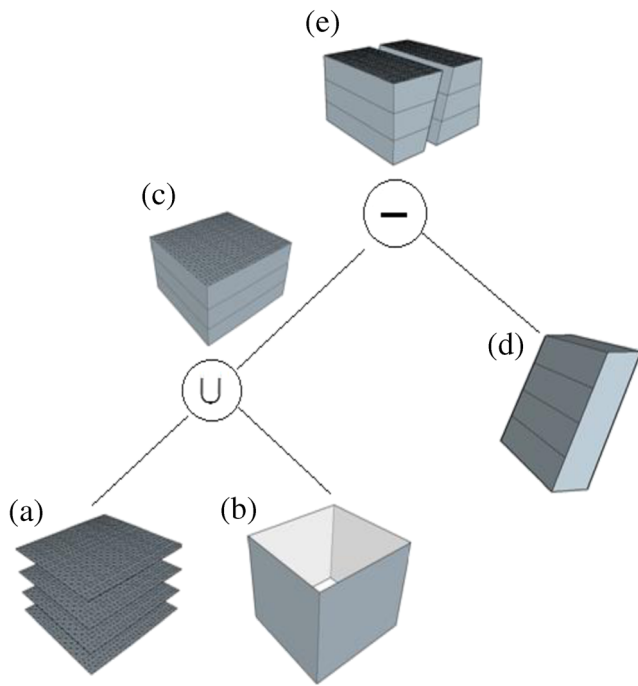
Fig. 1 Sketch of 3D modeling the complex geological bodies described with CSG

Fig. 2 The work flow of 3D modeling and visualizing complex geological bodies by combining ArcGIS and SketchUp

taken as the major consideration in the processes of modeling and visualization.

The modeling method not only can be operated automatically for creating the strata interfaces, but also can automatically deal with the strata intersections efficiently by the script tool implemented by combining the Python program language and Arcpy site package. In addition, it provides a semi-automatic approach to create the 3D modeling of complex geological bodies by the Boolean plug-ins of SketchUp. In addition, it can rapidly implement various types of visualization operations such as cutting, excavating and tunneling. The work flow includes five steps, as shown in Fig. 2.

Data acquisition, preprocessing and defining the modeling boundary

The modeling data are acquired from the geological exploration and survey for urban subway constructing projects, which mainly include various types of borehole data and geological maps. The borehole data are stored in different Excel files and can be classified into four types of borehole data such as geological, engineering, seismic and high-density. First, the DBF tables, such as basic borehole information, strata codes and strata lithology (Tables 1, 2 and 3), are extracted by displaying XY data and exporting files. Next, the borehole 3D points and 3D polylines (Fig. 3a and b) are created with the script tools implemented with the Python program language and the Arcpy site package (Fig. 4).
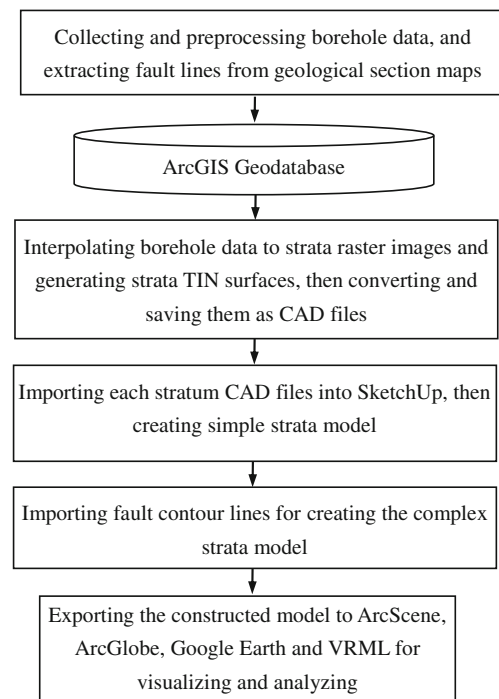
Then, the 2D polylines of faults are extracted from the geological maps with the automatic vectorization tool of ArcScan, and the convex polygon for defining the modeling boundary is extracted from the original borehole data with the tool of minimum bounding geometry (MBG), as shown in Fig. 5.

Furthermore, different types of modeling data, such as borehole 3D points, 2D fault lines and the 2D polygon of modeling boundary, are exported centrally into the GIS database, which will be used as the initial sample data for subsequent data preprocessing, modeling and visualization.

Finally, a series of virtual strata are added into borehole data respectively for dealing with the missing strata and local lenses (Fig. 5), and the zero thick values are used for replacing the missing values of every stratum in the borehole data. For example, the stratum Q2 is missing from the borehole data for boreholes BH10053 to BH10058. Thus, in order to construct a continuous stratum of Q2, all the thick values of the stratum

Table 1 Borehole code

ID	BH_CODE
1	D010001
2	D010002
3	D010003
4	D010004
…	…
131	D010131

Table 2 Strata code

ID	STRATA_CODE
1	Q4
2	Q3
3	Q2
4	N
5	FH
6	J
7	T1
8	T2
9	R5

Q2 among them are replaced with the zero thick values for dealing with the missing data.

Constructing a 3D model of complex geological bodies

Creating strata interfaces automatically

ModelBuilder is one of the most useful tools of ArcGIS, which can provide an automatic approach for creating the strata interfaces by designing a model that can be run automatically, and its operation steps mainly include: First, the 3D points of the borehole are inserted into ModelBuilder (Fig. 6a). Next, the iterator of feature selection and the tools of Arctoolbox such as "Select", "Kriging interpolation" (Fig. 6b), "RasterToPoint" (Fig. 6c), "CreateTin" (Fig. 6d), "TinTriangle" (Fig. 6e) and "TriangleToCAD" (Fig. 6f) are added and their generated intermediate files of the processes are shown in Table 4. Then, the borehole data, the iterator and tools are connected sequentially with the connection tool. Further, the input, output, workspace and other related parameters of the iterator and the tools are configured. Finally, the constructed model can be run automatically and create a series of DWG files of the strata interfaces after validating the constructed model; the designed model is shown in Fig. 6g.

It should be noted that there will be some intersections partially among the strata interfaces after the kriging interpolation (Fig. 7a). For example, the stratum interface of Q3 passes through the strata interfaces of Q2, N, FH, J-T1-T2 and R5. Here, the strata intersections among them are handled after converting the raster files of all strata into strata points. Meanwhile, the converted strata points are reprocessed with a Python script tool (Fig. 7c), and the heights of the strata such as Q2, N, FH, J-T1-T2 and R5 are accessed orderly and their heights are replaced with the values of above strata if their values are greater than them, The main implementing codes are shown as follows and the processed results are shown in Fig. 7b.

```
infcs = ConversionUtils.gp.GetParameterAsText(0)
infcs = ConversionUtils.SplitMultiInputs(infcs)
list1=[]
for infc in infcs:
    list1.append(infc)
l=len(list1)
for i in range(1,l-1,1):
    cursor1 = arcpy.da.SearchCursor(list1[i], ["SHAPE@XY","GRID_CODE","ID"])
for row1 in cursor1:
    sfid=str(row1[2])
    grid_code1 = row1[1]
    cursor2 = arcpy.da.SearchCursor(list1[i+1],["SHAPE@XY","GRID_CODE","ID"],'ID=\" + sfid + '\")
    for row2 in cursor2:
        grid_code2 = row2[1]
    if(grid_code1-grid_code2<0):
        cursor3 = arcpy.da.UpdateCursor(list1[i+1],["SHAPE@XY","GRID_CODE","ID"],'ID=\" + sfid + '\")
        for row3 in cursor3:
            row3[1]=grid_code1
            cursor3.updateRow(row3)
            if cursor3:
                del cursor3
```

Table 3 Strata lithology

ID	BH_CODE	TOP_ ELEVATION	BOTTOM_ ELEVATION	STRATA_ CODE	X	Y
1	10,053	9	−15	Q4	678,786	3,553,106
2	10,054	9	−13	Q4	679,954	3,553,317
3	10,055	10	−11	Q4	680,472	3,553,948
4	10,056	10	−11	Q4	681,579	3,555,069
5	10,057	11	−9	…	…	…
6	10,058	11	−7	Q4	678,749	3,554,881

Then, the reprocessed strata points are used as the input data of the model for creating the strata interfaces again.

Constructing the simple strata model

The Makefaces plug-ins and the tool of intersecting faces are combined for creating the simple strata semi-automatically, and its operation steps mainly include: First, the DWG files of strata interfaces created previously by ModelBuilder are sequentially imported into the visual modeling environment of SketchUp (Fig. 8a); Next, the strata surfaces and the boundary surface are created, respectively, with the polylines of strata interfaces and modeling boundary by the makefaces plug-ins (Fig. 8b). Then, the boundary polyhedrons are created by the push/pull tool, while the intersect lines between the strata surfaces and boundary polyhedron can be easily generated by the intersect faces tool. Further, the side surfaces of the modeling boundary are generated automatically by the explode tool, and the excess structure faces are erased for constructing the B-rep model of a single stratum. Last, the same operation method is adopted for constructing the B-rep models of other strata and composing them to the simple strata model, and different color materials are

assigned to stratum model respectively, as shown in Fig. 8f.

Creating the complex strata model with faults

The geometric shape and structure of a single fault can be described with a spatial curved surface and approximated with a complicated surfaces made up of a series of small facets (Wu and Xu 2003). Here, considering the feasibility and practicality of the modeling operation, the spatial planes are used for describing the faults surfaces since there are only the data of 2D fault polylines. Meanwhile, in order to intuitively represent their spatial distribution and topological relations in the 3D model of complex geological bodies, a series of polyhedrons are used for depicting the faults bodies. Initially, a series of polygons are generated from the 2D fault polylines by the buffer tool of ArctoolBox (Fig. 9a) and exported as the DWG file of the faults; Next, it is imported to SketchUp, and the surfaces of the faults are created automatically by the makefaces plug-ins; Then, an auxiliary line perpendicular to the surface of a fault surface is drawn manually; meanwhile, a spatial curve depicting its tendency is also drawn with a certain angle by the "Uniform B-Spline"

Fig. 3 Extracting and symbolizing of the borehole data: (**a**) 3D points, (**b**) 3D polylines

(a)
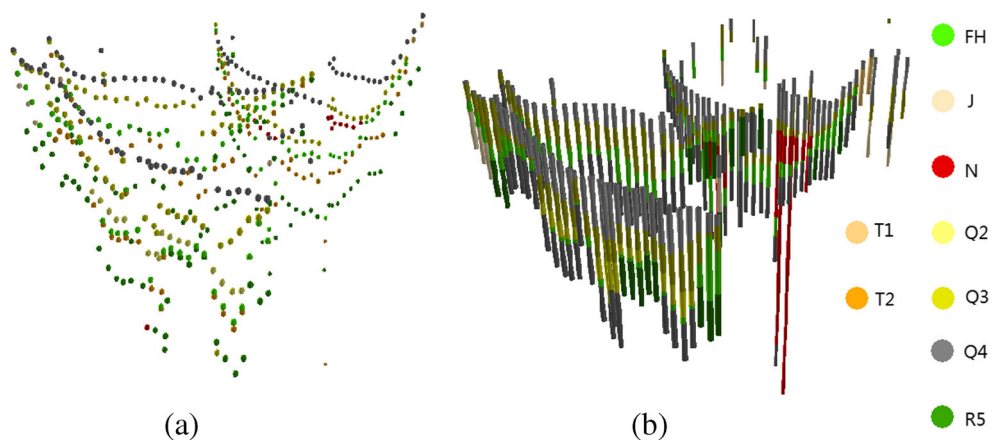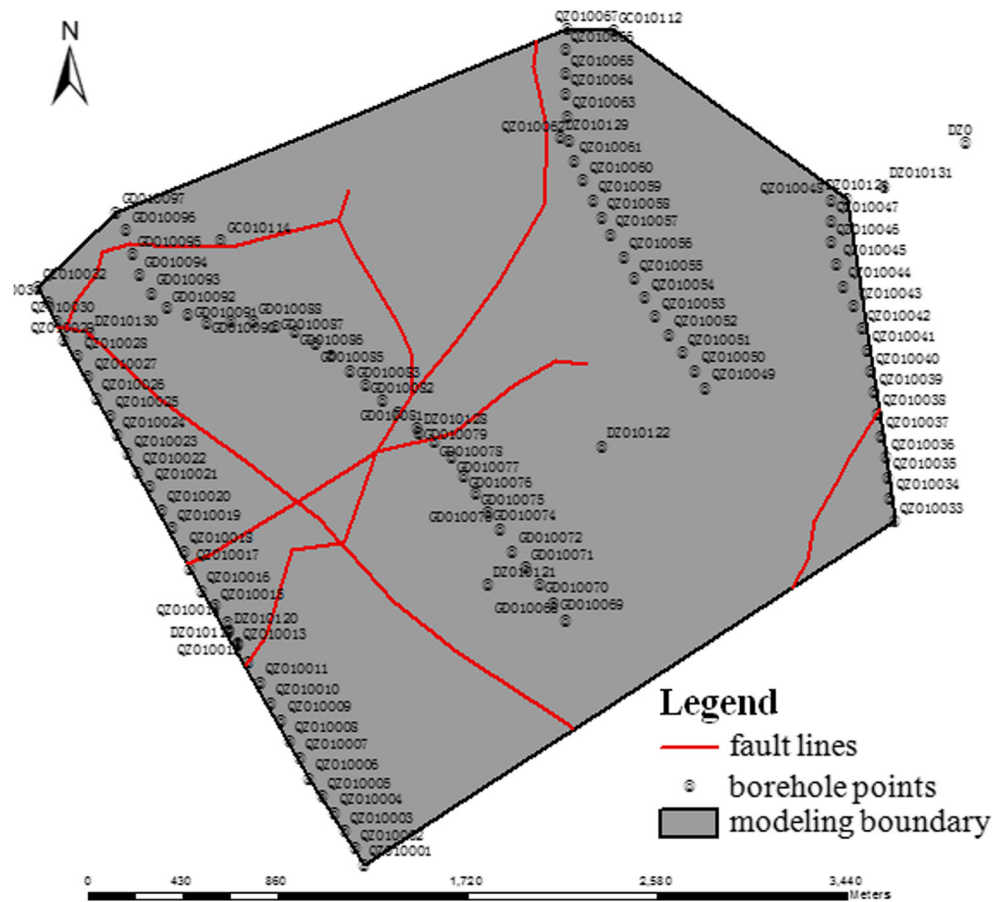
(b)

FH
J
N
T1   Q2
T2   Q3
      Q4
R5

Fig. 4 Extracting the convex
polygon of the modeling
boundary from the original
borehole data

plug-in along the auxiliary line (Fig. 9b). Furthermore, the polyhedron of the fault is drawn along the curves by the follow-me tool (Fig. 9c), and other faults polyhedrons are also created by the same operations (Fig. 9d) and embedded into the constructed simple strata model previously (Fig. 9e). Last, a series of Boolean operations between fault polyhedrons and the simple strata model are performed for generating the complex

strata model with faults by the Boolean plug-ins, as shown in Fig. 9f.

Operating and visualizing the constructed model

After constructing the complex strata models, some visualizing operation, such as cutting (Fig. 10a and Fig. 10b), excavating and tunneling (Fig. 10c and Fig. 10ed), can be easily achieved with the same Boolean operation by the Boolean plug-ins. Meanwhile, the constructed model can also be exported to ArcScene for further visualization and analysis (Fig. 10e). The detailed operations of these tools will not be introduced in this paper; users can easily learn them through the online plug-in courses of SketchUp.

Publishing and sharing the constructed model

So far, the constructed 3D strata model can only be operated in the environments of SkethUp and ArcScene. In order to publish and share the constructed 3D geological models to the public through the Internet, the users can easily access them by different
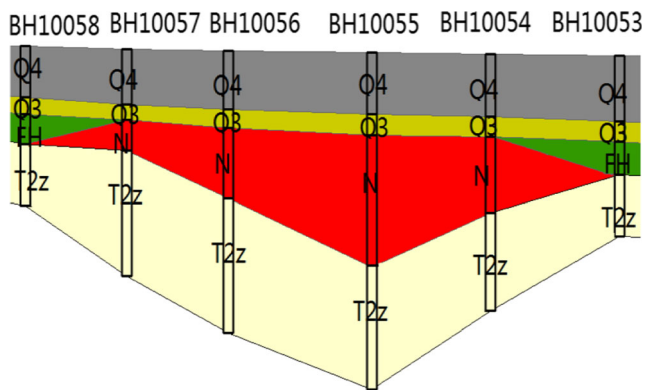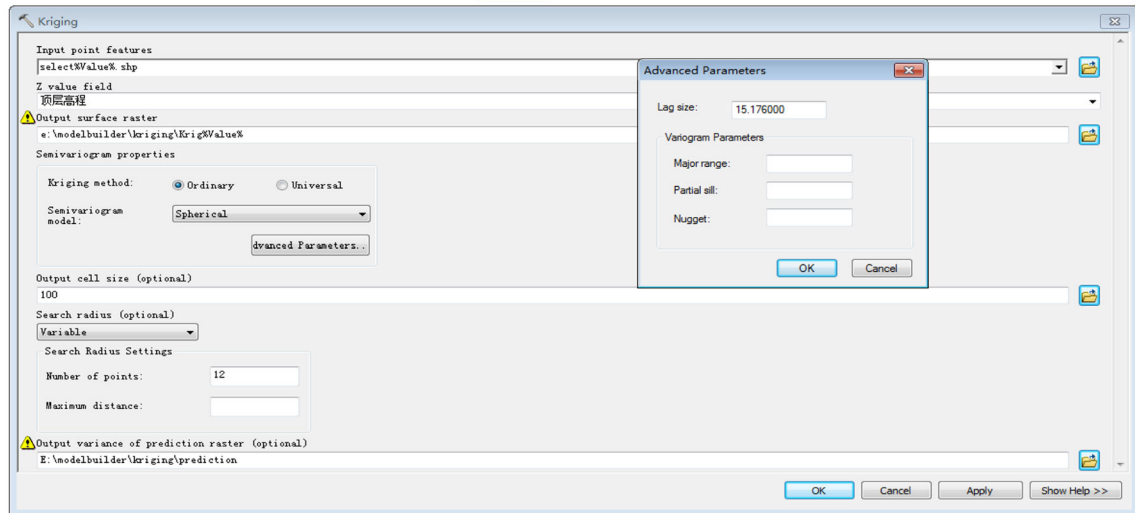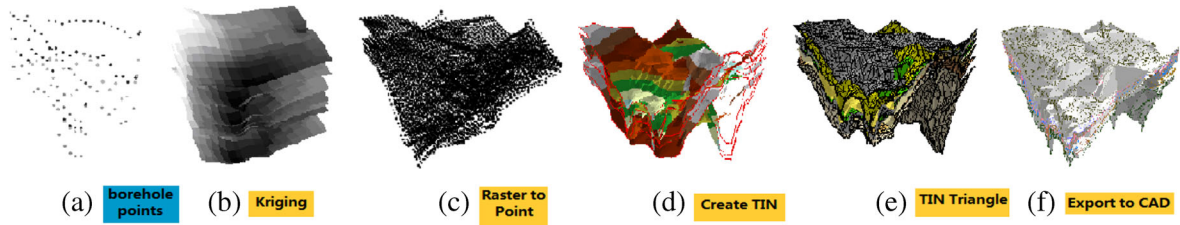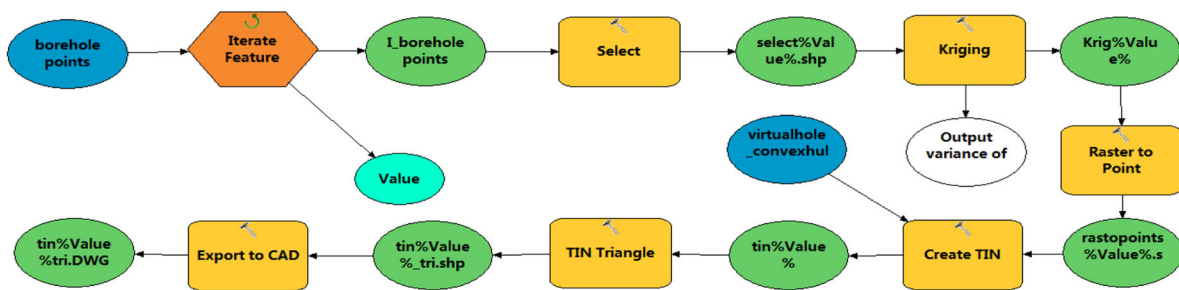

Fig. 5 Adding virtual strata to deal with the missing strata and local lenses in borehole data

(a) borehole points (b) Kriging (c) Raster to Point (d) Create TIN (e) TIN Triangle (f) Export to CAD

(g) Kriging interpolation and its parameter configuration

(h) The model designed by ModelBuilder

Fig. 6 The process and its results of automatically creating strata interfaces: (**a**) borehole 3D points, (**b**) the rasters of strata interfaces created by the Kriging interpolation tool, (**c**) the feature points created by RastertoPoint tool, (**d**) the tins of strata interfaces created by CreateTin tool, (**e**) the triangles of strata interfaces created by the TinTriangle tool, (**f**) the DWG polylines created by the ExportCAD tool, (**g**) kriging interpolation and its parameter configuration, and (**h**) the model designed by ModelBuilder

Table 4 The files generated during execution of the model

| Strata Id	File type	shp (point)	GRID	shp (point)	TIN	shp (polygon)	DWG (polyline)
1		strataq4	krigingq4	pointsq4	tinq4	Triangle4	Triangleq4
2		strataq3	krigingq3	pointsq3	tinq3	Triangleq3	Triangleq3
3		strataq2	krigingq2	pointsq2	tinq2	Triangleq2	Triangleq2
4		stratafh	krigingfh	pointsfh	tinfh	Trianglefh	Trianglefh
5		stratan	krigingn	pointsn	tinn	Trianglen	Trianglen
6		strataj	krigingj	pointsj	tinj	Trianglej	Trianglej
7		stratat1	krigingt1	pointst1	tint1	Trianglet1	Trianglet1
8		stratat2	krigingt2	pointst2	tint2	Trianglet2	Trianglet2
9		stratar5	kriging5	pointsr5	tinr5	Triangler5	Triangler5
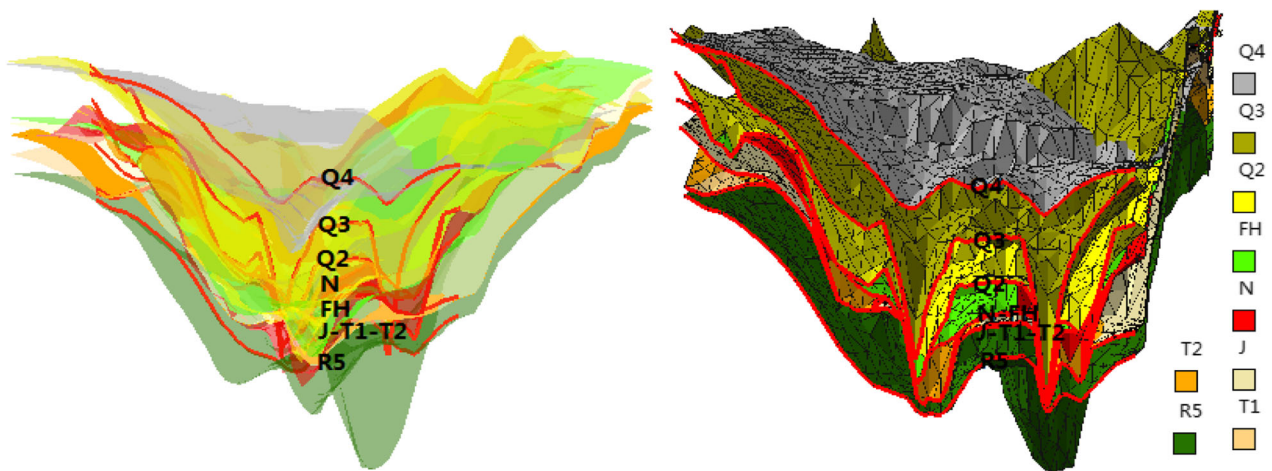
(a) rasters created by Kriging interpolation
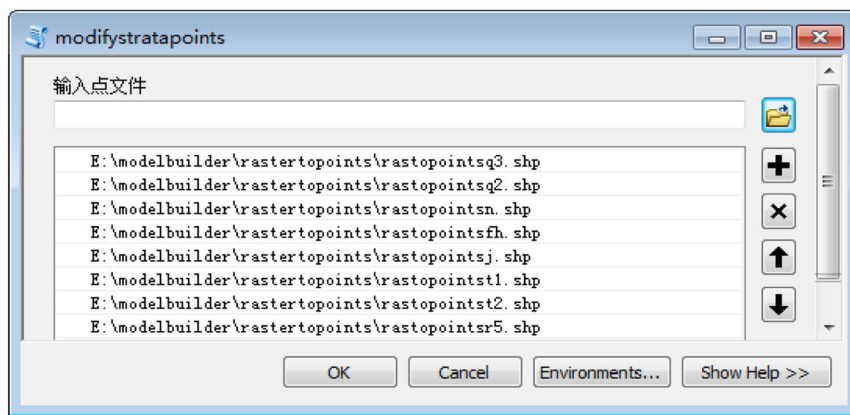
(b) strata triangles created by TIN triangles

(c) the python script tool for processing strata points
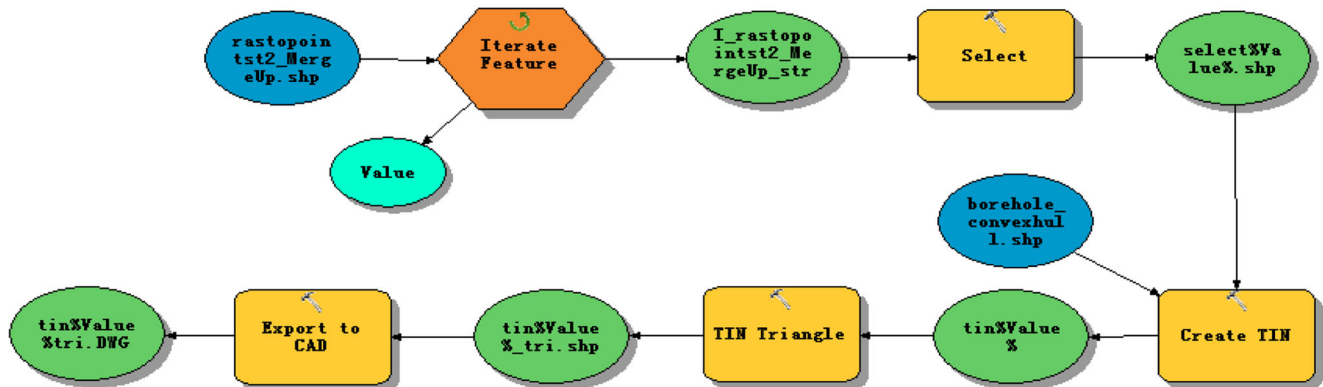
(d) the designed model for dealing with strata intersections

Fig. 7 The method for dealing with the intersections among strata interfaces: (**a**) raster created by kriging interpolation, (**b**) strata triangles created by TIN triangles, (**c**) the Python script tool for processing strata points, (**d**) the designed model for dealing with the strata intersections

Fig. 8 The processes and results of constructing the simple strata model in SketchUp: (**a**) the strata polylines of strata and modeling boundary imported from DWG files, (**b**) the strata surfaces created by the makefaces plug-ins, (**c**) acquired side faces by the intersect faces tool, (**d**) the constructed B-rep model of a single stratum by Boolean plug-ins, (**e**) the constructed simple strata model, (**f**) the simple strata model with colored materials
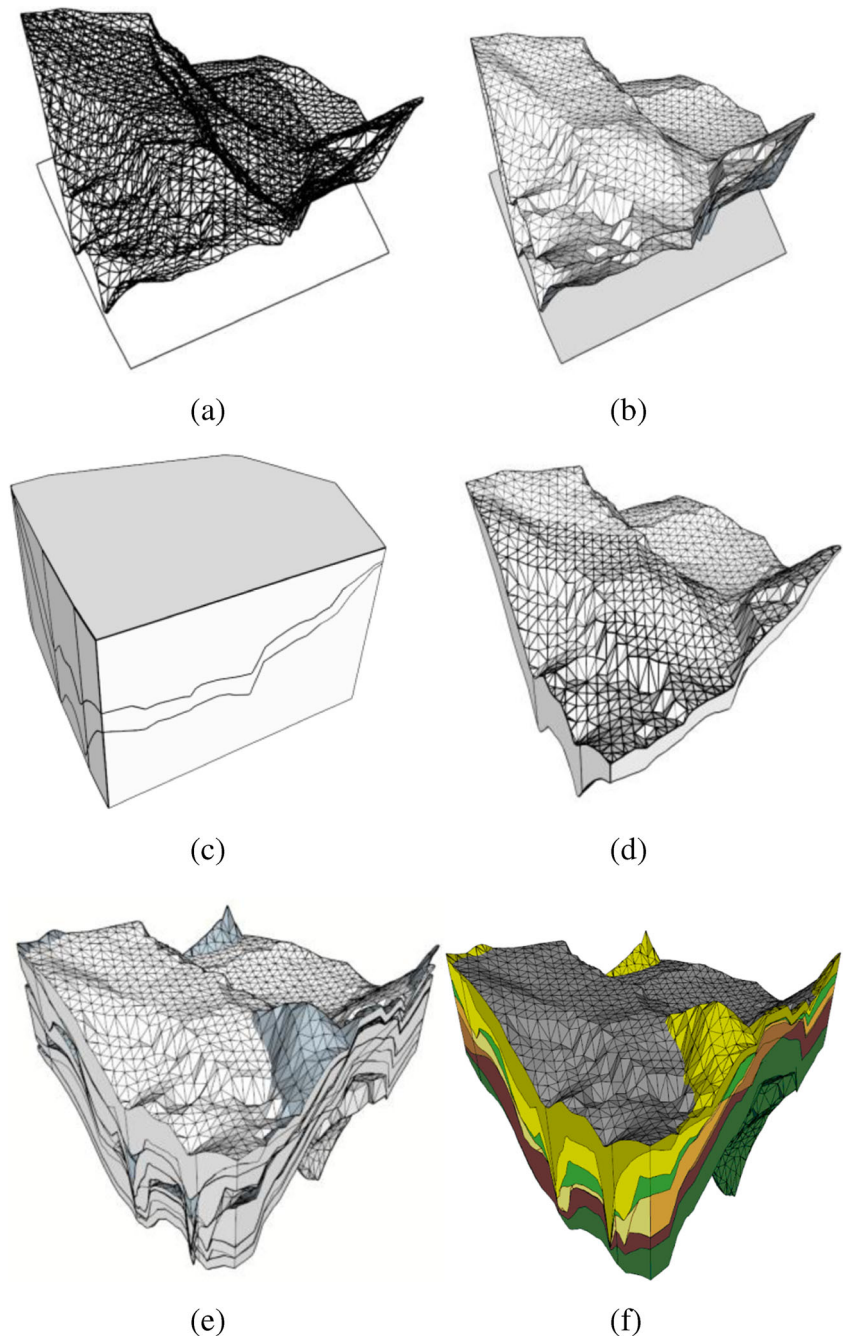

(a)

(b)

(c)

(d)

(e)

(f)

Fig. 9 The processes and its
results of constructing the
complex strata model with faults
by the Boolean plug-ins of
SkecthUp: (**a**) the faults polygons
generated by the buffer tool, (**b**)
the polyhedrons created by the
makefaces plug-ins and push-pull
tool, (**c**) a single fault polyhedron
and auxiliary surface drawn by
the drawing tool, (**d**) the fault
polyhedron slides at a certain
angle along the bottom edge of
the auxiliary surface, (**e**) the
simple strata model embedded
with fault polyhedrons, (**f**) the 3D
model of complex strata with
faults constructed by the Boolean
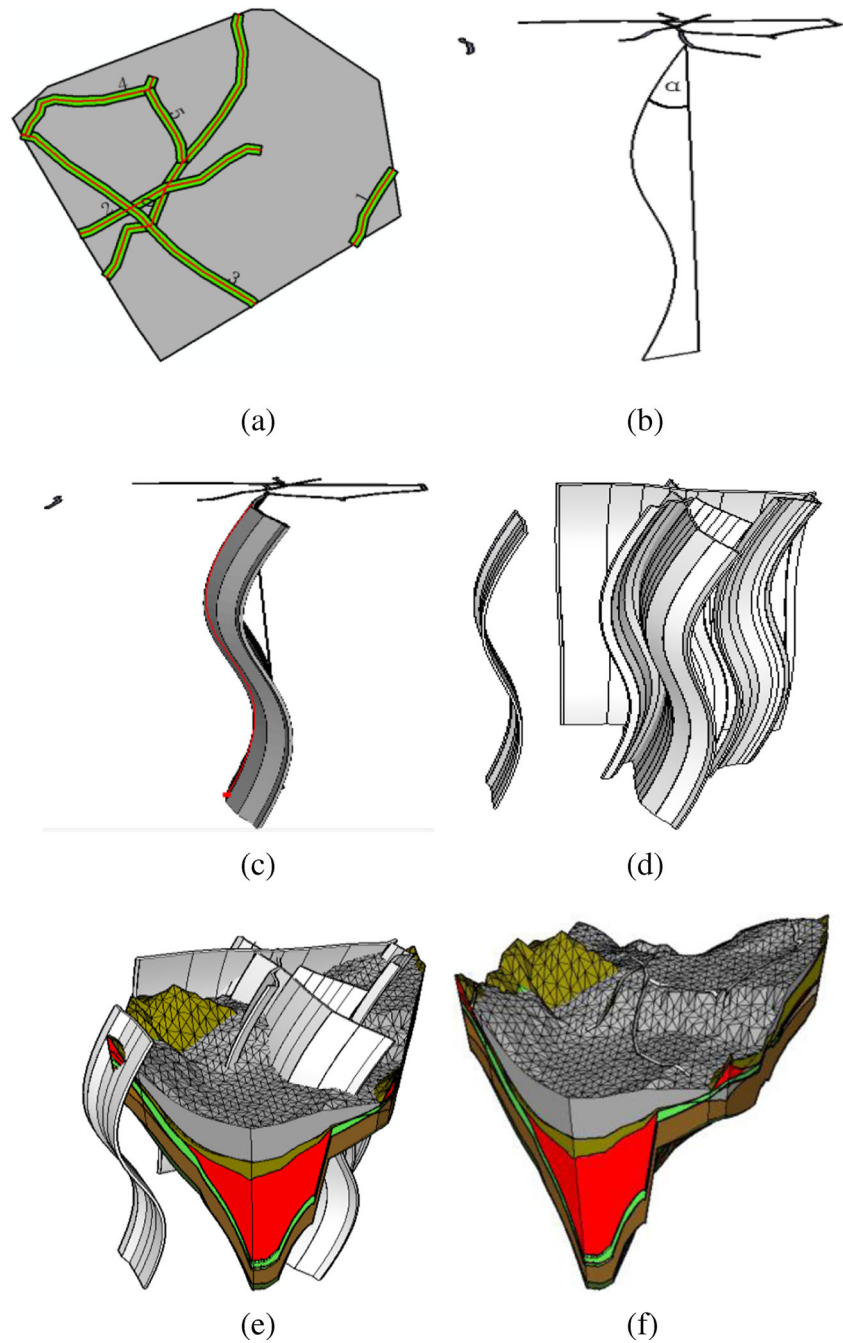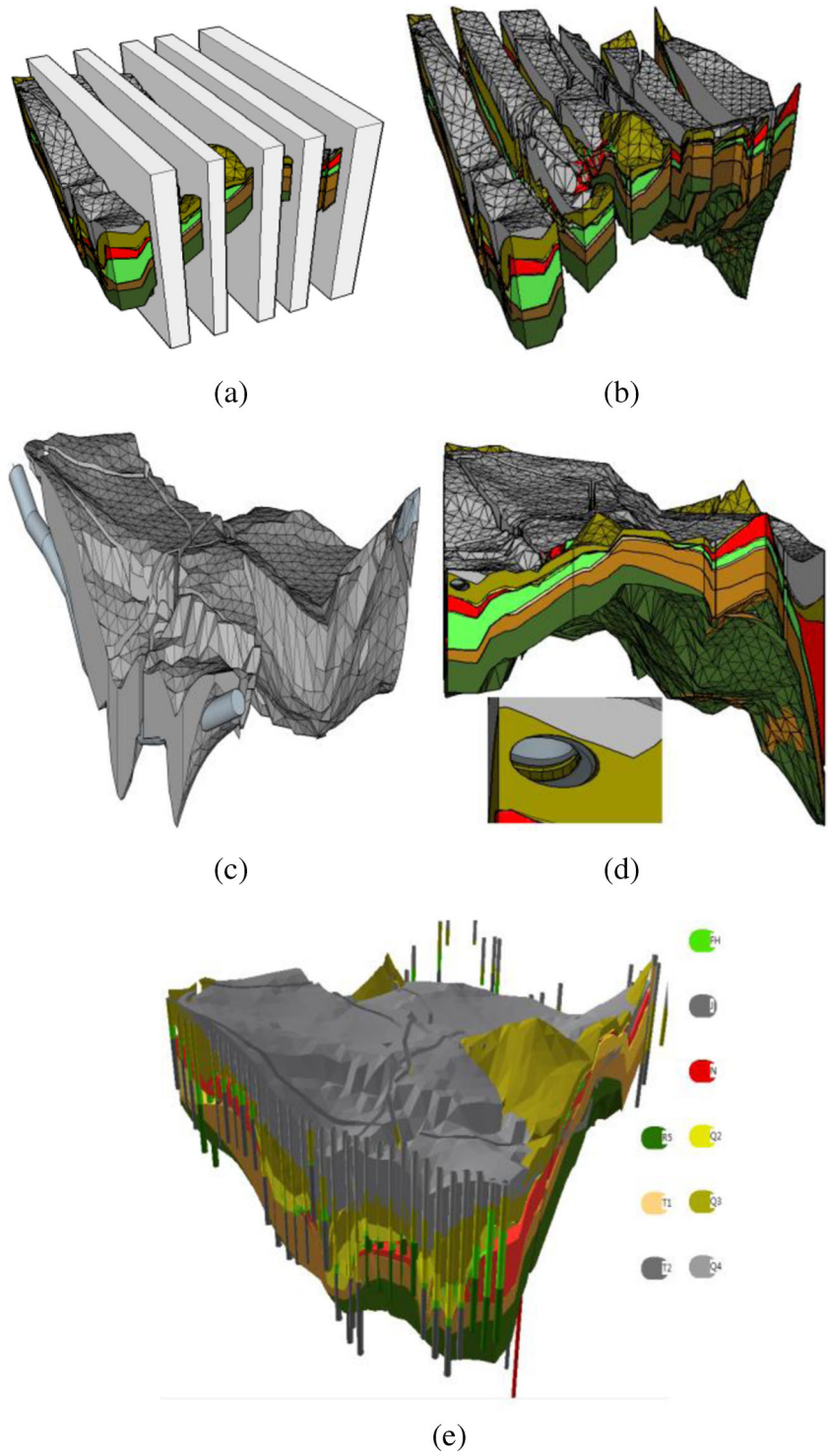plug-ins

(a)

(b)

(c)

(d)

(e)

(f)

Fig. 10 The processes and its results of operating and visualizing the constructed model by the Boolean plug-ins of SketchUp and the "Symbolization" tool of ArcScene: (**a**) and (**b**) vertical cutting on strata bodies, (**c**) and (**d**) tunneling a hole in strata bodies, (**e**) symbolizing the 3D polylines of borehole as tubes with strata bodies
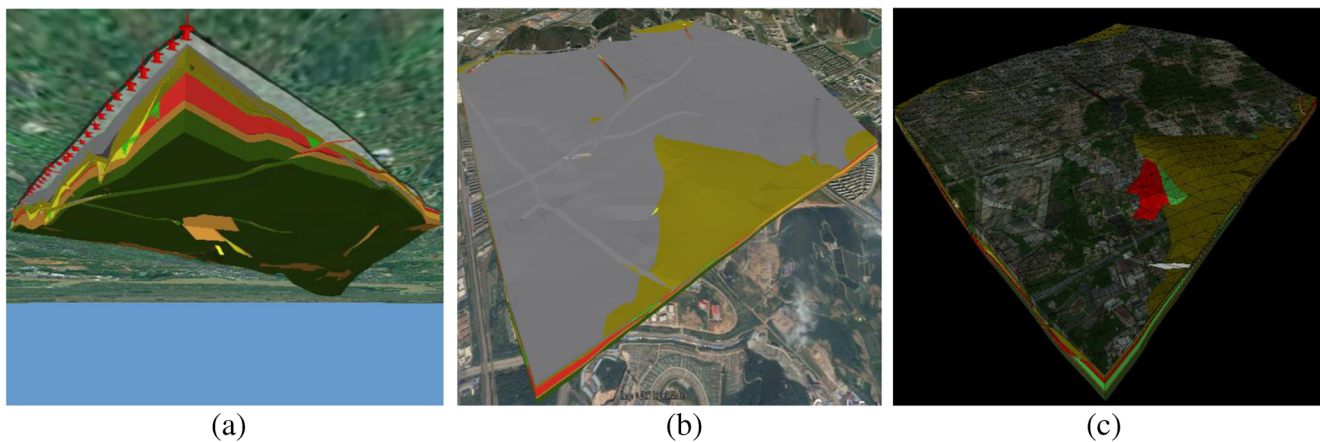

(a)

(b)

(c)

(d)

(e)

Fig. 11 Publishing and sharing the constructed 3D strata model: (**a**) in ArcGlobe, (**b**) in GoogleEarth, (**c**) in Internet Explorer by the BS Contact VRML plug-ins

types of client software such as ArcGlobe, GoogleEarth and Web browsers. The models can be exported as different file formats such as 3DS, COLLADA and VRML for further publishing and sharing. The exported 3D models in the above client software are shown in Fig. 11.

Conclusions and outlook

Many effective modeling methods and some commercial modeling software have been proposed and implemented since the technology of 3D geological modeling was born. The modeling method proposed in this paper combines the merits of a typical GIS system and 3D modeling software such as ArcGIS and SketchUp for semi-automatically 3D modeling and visualizing the complex geological bodies, and it has the following advantages:

(1) ArcGIS is one of the most widely used GIS systems, which can effectively help to manage, process and analyze the multi-source geological data based on its powerful functions such as data processing, mapping and visualization since many technicians have been accustomed to the thinking and operation habits of GIS systems.

(2) The Python program language, Arcpy site package and the visual modeling tool of ModelBuilder are combined for data preprocessing and modeling, which not only can greatly improve the efficiency of data preprocessing and modeling, but also provide more operational stability and flexibility.

(3) SketchUp is simple and practical 3D modeling software which can greatly help to construct complex geological bodies intuitively by using its easy drawing functions and rich plug-in tools, and implement various types of modeling operations; Meanwhile, the constructed strata models can be easily exported to various file formats such as 3DS, COLLADA, VRML, etc.

(4) ArcGIS and SketchUp are combined for 3D modeling and visualizing complex geological bodies, which not only can effectively help to improve the efficiency of data preprocessing, modeling and visualization, but also can reduce the labor intensity of technicians, and is very beneficial to promote the popularization of 3D geological modeling technology in the fields of mining, geology and engineering construction.

(5) There are still many difficulties for 3D modeling and visualizing complex geological bodies automatically due to the variety of geological settings and data preprocessing. The actual modeling work not only requires automatic processes for data preprocessing, modeling and visualization, but also the constructed models have to be modified manually for making them close to reality.

This paper proposes a semi-automatic method for modeling and visualizing complex geological bodies by combining ArcGIS and SketchUp, and its main purpose is to explore a simple and practical 3D geological modeling method for promoting the further popularization in the fields of mining, geology and engineering constructions widely. The modeling method described in this paper has made modeling experiments on simple stratified geological bodies with the most common formations such as faults and missing stratum. Special complex geological bodies, such as deformed and reversed bodies, has not been studied and will be the emphasis of future research work.

References

Alan ML, Norman LJ (2003) Building solid models from boreholes and user-defined cross-sections. Comput Geosci 29(5):547–555

Cheng PG, Wang CG, Gan WJ et al (2005) A hybrid 3D data model based on multi-DEMs and QTPVs and its application in geological modeling. J Jilin Univ (Earth Sci Ed) 35(6):806–811

Choi Y, Yoon SY, Park HD (2009) Tunneling analyst: a 3D GIS extension for rock mass classification and fault zone analysis in tunneling. Comput Geosci 35(6):1322–1333

Galera C, Bennis C, Moretti I, Mallet JL (2003) Construction of coherent 3D geological blocks. Comput Geosci 29(8):971–984

Gong JY, Cheng PG, Wang YD (2004) Three-dimensional modeling and application in geological exploration engineering. Comput Geosci 30(4):391–404

Hou WS, Yang L, Deng DC et al (2016) Assessing quality of urban underground spaces by coupling 3D geological models: the case study of Foshan city, South China. Comput Geosci 89:1–11

Houlding SW (1994) 3D geoscience modeling, computer techniques for geological characterization. Springer, New York

Kaufmann O, Martin T (2008) 3D geological modelling from boreholes, cross-sections and geological maps, application over former natural gas storages in coal mines. Comput Geosci 34(1):70–823

Lázaro JM, Navarro JÁS, Gil AG et al (2014) 3D-geological structures with digital elevation models using GPU programming. Comput Geosci 70(C):138–146

Li PJ (2000) Three dimensional modeling and visualization for stratified geological objects. Earth Sci Front 7(S):271–277

Mallet JL (2002) Geomodeling. Oxford University Press, New York

Martin-Izard A, Arias D, Arias M et al (2015) A new 3D geological model and interpretation of structural evolution of the world-class Rio Tinto VMS deposit, Iberian Pyrite Belt (Spain). Ore Geol Rev 71:457–476

Mccarthy JD, Graniero PA (2006) A GIS-based borehole data management and 3D visualization system. Comput Geosci 32(10):1699–1708

Tonini A, Guastaldi E, Massa G et al (2008) 3D geo-mapping based on surface data for preliminary study of underground works: a case study in Val Topina (Central Italy). Eng Geol 99(1):61–69

Wang GW, Huang LT (2012) 3D geological modeling for mineral resource assessment of the Tongshan cu deposit, Heilongjiang Province, China. Geosci Front 03(4):483–491

Wang BJ, Shi B, Song Z (2009) A simple approach to 3D geological modelling and visualization. Bull Eng Geol Environ 68(4):559–565

Wu LX (2004) Topological relations embodied in a generalized tri-prism (GTP) model for a 3D geoscience modeling system. Comput Geosci 30(4):405–418

Wu Q, Xu H (2003) An approach to computer modeling and visualization of geological faults in 3D. Comput Geosci 29(4):503–509

Wu Q, Xu H, Zou XK (2005) An effective method for 3D geological modeling with multi-source data integration. Comput Geosci 31(1):35–43

Wu LX, Zhang RX, Qi YX et al (2002) 3D geoscience modelling and virtual mine system. Acta Geodaet Cartograph Sin 31(1):28–33 (in Chinese)

Xiao K, Nan L, Alok P, Holden E-J, Bagas L, Lu Y (2015) GIS-based 3D prospectivity mapping: a case study of Jiama copper-polymetallic deposit in Tibet, China. Ore Geology Reviews 71(3):611–632.

Xu NX, Tian H (2009) Wire frame: a reliable approach to build sealed engineering geological models. Comput Geosci 35(8):1582–1591

Xu NX, Tian H, Kulatilake PHSW et al (2011) Building a three dimensional sealed geological model to use in numerical stress analysis software: a case study for a dam site. Comput Geotech 38(8):1022–1030

Yang XD, Hu LT, Tang ZH (2006) 3D modelling and visualization of stratum based on java/java 3D. Acta Geodaet Cartograph Sin 35(2):166–170 (in Chinese)

Zanchi A, Francesca S, Stefano Z et al (2009) 3D reconstruction of complex geological bodies: examples from the alps. Comput Geosci 35(1):49–69

Zhu LF, Pan X, Wu XC (2006) Design and development of 3D geological modeling and visualization system. Rock Soil Mech (27, 5):829–831 (in Chinese)

Zhu LF, Zhang C, Li M et al (2012) Building 3D solid models of sedimentary stratigraphic systems from borehole data: an automatic method and case studies. Eng Geol 127(1):1–13