



Heriot-Watt University
Research Gateway

A conditional GAN-based approach to build 3D facies models sequentially upwards

Citation for published version:

Sun, C, Demyanov, V & Arnold, D 2023, 'A conditional GAN-based approach to build 3D facies models sequentially upwards', *Computers and Geosciences*, vol. 181, 105460.
<https://doi.org/10.1016/j.cageo.2023.105460>

Digital Object Identifier (DOI):

[10.1016/j.cageo.2023.105460](https://doi.org/10.1016/j.cageo.2023.105460)

Link:

[Link to publication record in Heriot-Watt Research Portal](#)

Document Version:

Publisher's PDF, also known as Version of record

Published In:

Computers and Geosciences

Publisher Rights Statement:

© 2023 The Author(s).

General rights

Copyright for the publications made accessible via Heriot-Watt Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

Heriot-Watt University has made every reasonable effort to ensure that the content in Heriot-Watt Research Portal complies with UK legislation. If you believe that the public display of this file breaches copyright please contact open.access@hw.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



Research paper

A conditional GAN-based approach to build 3D facies models sequentially upwards

Chao Sun^{*}, Vasily Demyanov, Daniel Arnold

Institute of Geoenery Engineering, Heriot-Watt University, Edinburgh, EH14 4AS, Scotland, UK¹

ARTICLE INFO

Keywords:

Generative adversarial networks
3D meandering facies model
3D reconstruction

ABSTRACT

This paper presents an alternative way of simulating 3D facies models using conditional generative adversarial networks (GANs) by reconstructing the 3D volume upward with customised thickness from generated 2D slices. To mimic natural fluvial reservoirs, GANs need to learn the complicated facies' lateral and vertical sequences, connectivity patterns and spatial correlation, e.g. the distribution of channel deposits and lateral accretion packages, including channel lags and point bars in a meandering fluvial system. 3D GAN modelling remains challenging, despite some success in 3D, due to the increased data and GAN model complexity caused by the 3D training data. For example, a 3D facies model with a larger number of voxels introduces more complex geobody shapes and requires 3D convolutions that increases the computational burden compared to 2D convolutions.

Our proposed conditional GAN-based framework predicts 2D upper layers slice by slice to build 3D facies models, mimicking the sedimentary process in a purely aggrading system, which is less computationally costly than generating the facies models used as training data and generating facies models directly in 3D from a GAN. Each upper layer is predicted conditionally to a lower layer that contains the facies distributions in all layers below the target using several unique training enhancements. All those methods help prevent the simulated 3D models from gradually losing realism and the preset sedimentary settings vertically while inferring the plausible evolution of the channel system. The results demonstrate that the trained conditional generator, a SPADE generator, can work as a simulator to produce unconditional 3D meandering fluvial facies models by inputting a user-defined spatial correlation strength between layers and the number of layers needed (thickness in metres) without post-processing or retraining. This framework's performance was evaluated using a connectivity metric towards the range of avulsion scenarios that represent a range of depositional uncertainty.

1. Introduction

Recent publications proved that General Adversarial Networks (GAN) could successfully learn complex geological patterns, for example, fluvial channels from a set of object-based 2D and 3D facies realisations (Laloy et al., 2018; Chan and Elsheikh, 2019; Zhang et al., 2019; Song et al., 2021b,a, 2022), but few pieces of research focus on learning 3D models from process-based models. Most studies used object-based models to test GANs' performance in learning pre-defined shapes, while natural depositional systems can produce more complicated patterns. For example, meandering systems often create fluvial channels migrating across the field, eroding the outer bank and leaving sand and lag deposits at the inner bank. With the meander belt evolving, different types of channel abandonment occur, such as avulsion and meander cutoffs, filling old channels with mud (or some sand at the bottom) and developing new channels (Heller and Paola, 1996; Slingerland and Smith, 2004; Russell, 2017). Process-based models, therefore,

can provide more plausible realisations than object-based methods by mimicking those sedimentary processes (Lopez, 2003; Grimaud et al., 2022). In Sun et al. (2023), we highlighted GANs' learning difficulties when reproducing process-based models in 2D and developed a new GAN variant with three improvements to prevent creating implausible features, named Fluvial GAN. (1). Fluvial GAN uses indicator variables to encode categorical data such as rock type. (2). Fluvial GAN uses a Hybrid-discriminator based on the PatchGAN discriminator and dilated convolution to jointly penalise the data structure in different sizes of receptive fields (input data region connected to the neurons). This empirical discriminator architecture helps prevent GAN from producing an unrealistic structure with plausible local features. (3). Fluvial GAN combines hinge loss and zero-centred gradient penalty in the loss function to reduce the risk of mode collapse, a common issue during GAN training where only part of training data are reproduced by the generator. However, applying GANs to 3D modelling remains

^{*} Corresponding author.

E-mail address: cs157@hw.ac.uk (C. Sun).

<https://doi.org/10.1016/j.cageo.2023.105460>

Received 28 February 2023; Received in revised form 15 September 2023; Accepted 19 September 2023

Available online 22 September 2023

0098-3004/© 2023 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

challenging and has a few drawbacks, impeding its application to real fields.

The computational burden is one of the challenges in GAN 3D modelling applications caused by 3D training data. Earlier studies extended GAN applications to produce 3D realisations by directly feeding 3D data into the GAN training framework, e.g., Laloy et al. (2018), Zhang et al. (2019) and Song et al. (2022). The size (in terms of the number of pixels/voxels) of 3D data is several times more voluminous than the 2D data used in previous GAN studies, making it hard for GAN realisations to match the more complex 3D patterns represented by the training data. This volume (in terms of the number of pixels/voxels) may increase further when dealing with multiple facies. For instance, if the GAN uses the One-Hot Encoder to process the facies categorical data, the data volume expands several times equal to the number of categories (Sun et al., 2023), further exacerbating the problems with pattern representation. Regarding the model architecture of GAN for 3D data, the kernel size in the 3D convolutional neural network (CNN) is often larger than in the 2D CNN because the 3D CNN's kernel has three spatial dimensions rather than two. For example, in 2D CNN, a single 3×3 kernel has 9 variables; in 3D CNN, one $3 \times 3 \times z$ kernel has $9 \times z$ variables, where z is an arbitrary value of the kernel size in the z -direction. Therefore, the GAN model gets more learnable parameters, impacting the convergence and stability of the gradient-based optimisation (Brock et al., 2018).

One approach to improve GAN performance in 3D is to add conditioning data that controls the spatial arrangements of facies. Yang et al. (2022) presented an example of using conditional GAN to reconstruct 3D samples from 2D cross-sections. They extracted 2D cross-sections from the GAN's 3D training data. The 2D cross-sections are then used to condition GAN's prediction of the rest of the 3D volume. With the help of conditioning data, GANs could match the reference data with good quality. While this conditional GAN approach still needs to process 3D data requiring big heavy GAN model structure with many learnable parameters.

Another challenge is that we cannot easily apply a GAN pre-trained on fluvial (or any particular depositional environment) data to a new field of a similar depositional setting but different reservoir sizes unless re-training the (whole or partial) GAN model or extending the latent vector if the generator is fully CNN design. Two time-consuming procedures of deep learning-based approaches are data preparation and model training. Therefore, reusing the pre-trained model effectively benefits the GAN application from modelling one field to another with the same sedimentary environments. For GANs with fully connected layers (Chan and Elsheikh, 2019), the output is fixed to the same size as the input training data. Chan and Elsheikh (2019) showed that extending the latent vector could help reuse the pre-trained GAN for creating larger 2D images. For GANs without fully connected layers (Laloy et al., 2018; Song et al., 2022), the output size can vary but only by maintaining the ratio of input latent cubes to the output facies models. In both cases, this limits the ability to apply GANs to reservoirs with different sizes and aspect ratios.

This paper introduces a novel use of a conditional GAN-based reconstruction approach, named FluvialGAN_3DR, for 3D geological facies modelling of meandering fluvial systems with different reservoir thicknesses without re-training. This reconstruction approach is trained to build 3D facies models iteratively layer by layer from bottom to top, conditioning each new layer on the layers below it. By that, we can control the number of layers in a facies model and, therefore, reservoir thickness in unit thickness.

Our training dataset, GAN River-I (Sun et al., 2022), is an ensemble of low NTG meandering fluvial models simulated by a processes-based model, *FLUMYT^M* (Lopez, 2003; Grimaud et al., 2022). Two reasons account for choosing a low NTG dataset. First, in order to see the associations between the channel, point bar and other facies, we need to have a lower NTG model as the sand bodies start to amalgamate into more sheet-like deposits in high NTG models. These high NTG

models also pose an interesting problem for characterisation but are harder to identify as being 'true' by eyeball. Secondly, we picked an NTG of around the percolation threshold to get connected sands and allow flow, though arguably, such a low NTG could be quite marginal volumetrically. There are fields with meandering rivers in them (Snieder et al., 2021; Villamizar et al., 2015)

We present two unique improvements to FluvialGAN_3DR training to prevent the reconstruction from gradually losing geological realism as the layers are added. One incorporates multiple reconstruction steps in training, and the other concatenates previously simulated layers with a decay mechanism to enhance the spatial correlation between the newly generated upper layer and generation history.

We organise the rest of the paper as follows:

- Section 2 briefly reviews GAN principles, introduces the conditional GAN generator used in this study, and describes our 3D modelling approach and two training enhancements.
- Section 3 presents the results showing the proposed methods lessen the prevalence of identified learning difficulties in geological realism and 3D pattern diversity.
- In Section 4, we compare our proposed methods and other approaches and discuss the pros and cons of the proposed conditional GAN-based framework in learning the 3D meandering fluvial models.
- Section 5 contains the paper's conclusions.

2. Methodology

This paper uses many terms from deep-learning research and specific names and notations throughout our work. We summarise those expressions in Table 1.

2.1. Generative adversarial networks (GAN)

In 2014, Goodfellow invented a deep generative model learning competitively between two neural networks called generative adversarial network (GAN) (Goodfellow et al., 2014). A GAN attempts to optimise the two neural networks, one called generator and the other one called discriminator, by reducing the errors from the loss functions of both models during training (see Fig. 1 for details of how a GAN works). This competitive mechanism makes GANs hard to train since the error reduction of one neural network is at the cost of increasing the other neural network error. Therefore, many variants aim to improve GAN performance and stabilise its training process (Radford et al., 2015; Arjovsky et al., 2017; Gulrajani et al., 2017; Lim and Ye, 2017).

2.2. Conditional GAN and SPADE generator

A conditional generative adversarial network is a branch of GANs that blends auxiliary information into its generations by incorporating conditioning data into the training process. Mirza and Osindero (2014) developed a conditional GAN model that adds conditioning data as an extra input to both the generator and the discriminator. Later, more publications directly used image maps to condition GAN generation by encoding the conditioning data to the latent vector or features maps, for example, *pix2pix* and *pix2pixHD* model (Isola et al., 2017; Wang et al., 2018).

Park et al. (2019) proposed a spatially-adaptive denormalisation (SPADE) generator to replace the encoder-decoder structure in *pix2pixHD* for semantic image synthesis. The SPADE generator down-samples the conditioning data to the same spatial size (e.g. in terms of the width and height for 2D images) as every hidden layer block and uses a conditional normalisation, called SPADE, to involve the conditioning information into its generations (see Fig. A.1 in Appendix A). The SPADE generator consists of several residual blocks of neural networks called SPADE ResBlk (see Fig. A.1 in Appendix A),

Table 1
Glossary of expressions in this paper.

Terms	Description
GAN	Generative adversarial networks.
SPADE	Spatially-adaptive denormalisation is a type of conditional normalisation.
CNN	Convolutional neural networks.
NTG	The net to gross ratio, this study refers to the proportion of sand-prone facies in a realisation.
FLUMY	A process-based channelised reservoir modelling software.
GAN River-I	Our raw dataset produced by FLUMY that is used to create the training dataset for GANs.
Fluvial GAN	Our GAN for 2D facies modelling, published in Sun et al. (2023).
FluvialGAN_3DR	Our proposed conditional GAN-based 3D reconstruction framework in this paper.
Fluvial GAN 3D	Our 3D facies modelling tool, composed of pre-trained Fluvial GAN and FluvialGAN_3DR.
Base layer	Refers to the initial layer in a 3D simulation, a 2D slice from Fluvial GAN or FLUMY.
Lower layer	Refers to the conditioning data at the current reconstruction iteration.
Upper layer	Refers to the target at the current reconstruction iteration.
Reconstruction step	One 3D reconstruction starting from a base layer operated in one training iteration.
Reconstruction iteration	Predict an upper layer based on the current lower layer in one reconstruction step.
Avulsion rate factor (N_{av})	A parameter used to calculate the conditioning data.

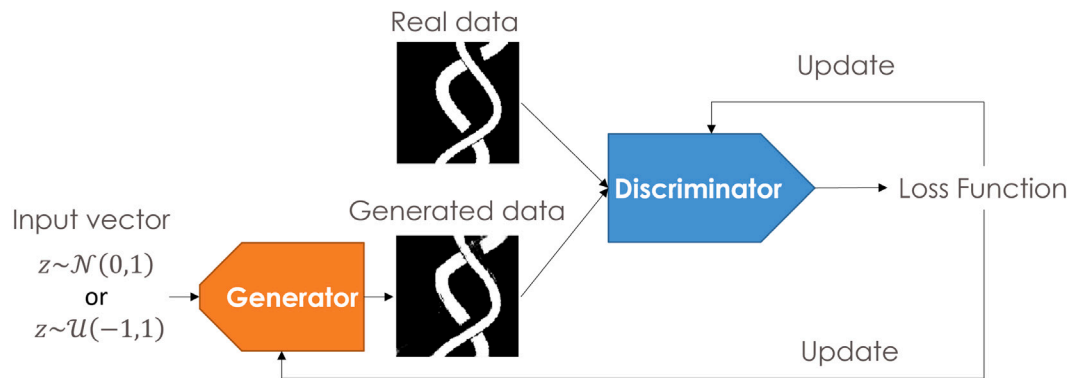


Fig. 1. A GAN workflow for modelling channelised patterns. The input vector refers to a set of elements obeying a pre-defined distribution, such as the uniform or normal distribution. The generator is the neural network learned to project an input vector to a realisation (generated data). The discriminator is the neural network trained to distinguish the generated data from the real data from the training dataset. The loss function calculates an error based on which neural network is to be updated and the origin of the data. Source: From Sun et al. (2023).

which is also the main neural network block used in our work. This SPADE generator has fewer trainable parameters than $pix2pixHD$, by around half, which benefits the training process and reduces the risk of overfitting (Park et al., 2019).

Spatially-adaptive denormalisation (SPADE) is a conditional normalisation that uses conditioning data to denormalise the output of an unconditional normalisation by using convolutional neural networks (Park et al., 2019). The convolutional neural networks in SPADE learn to produce a scaling tensor and a bias tensor to modulate every corresponding element in the normalised activations (see Fig. A.2 in Appendix A). The two learned parameters denormalise the element values by Eq. (1) (Park et al., 2019)

$$\gamma_{c,y,x}^i(m) \frac{h_{n,c,y,x}^i - \mu_c^i}{\sigma_c^i} + \beta_{c,y,x}^i(m) \quad (1)$$

where γ is the learned scaling parameter, β is the learned bias parameter, m is the conditioning data, h is the input of the normalisation, μ is the mean value of activation in a certain pixel set, σ is the standard deviation of activation in a certain pixel set, i denotes the layer number of a deep convolutional network, and n , c , y , and x are the index of the batch, channel, height, and width dimension respectively.

2.3. Fluvial GAN 3D modelling

This study presents a GAN-based approach for 3D facies modelling, which allows the modeller to control the reservoir thickness and adjust the 3D pattern reflecting different sedimentary settings (different avulsion rates in a purely aggrading system). The 3D simulation of the meandering facies models relies on a pre-trained conditional GAN generator, SPADE generator (Park et al., 2019). The program feeds an

initial *base layer* as the conditioning data to the pre-trained SPADE generator to predict its *upper layer* and then replaces the old conditioning data with the generated *upper layer* for simulating the next *upper layer* till it reaches the number of layers/metres required (see Fig. 2). We can either use the pre-trained Fluvial GAN to produce an initial *base layer* or pick a random 2D horizontal slice from FLUMY simulations. Then, we only need to set up two parameters for the reconstruction process: one is the number of layers reconstructed, and the other one, called the avulsion rate factor N_{av} , is optional. The avulsion rate factor controls the correlation between the current generation and earlier generations. We introduce the avulsion rate factor in detail as a training enhancement in Section 2.3.3.

To make the SPADE generator properly predict the *upper layer*, we use a conditional GAN training framework and propose two training enhancements to improve reconstruction stability.

- (1) A conditional GAN framework realises the 3D extension of Fluvial GAN, named FluvialGAN_3DR, by using the SPADE Generator to produce upper-layer realisations conditioned to given lower-layer realisations.
- (2) The multi-step enhancement prevents the predicted upper-layer realisations from fully losing realism by involving different iterations of the reconstruction upwards in the training process (see Section 2.3.2).
- (3) The conditioning data decay enhancement improves the diversity of the 3D facies model's top part (ten layers/metres above the base layer) by adding all layers below the current target to the conditioning data (*lower layer*) in a decay mechanism (see Section 2.3.3).

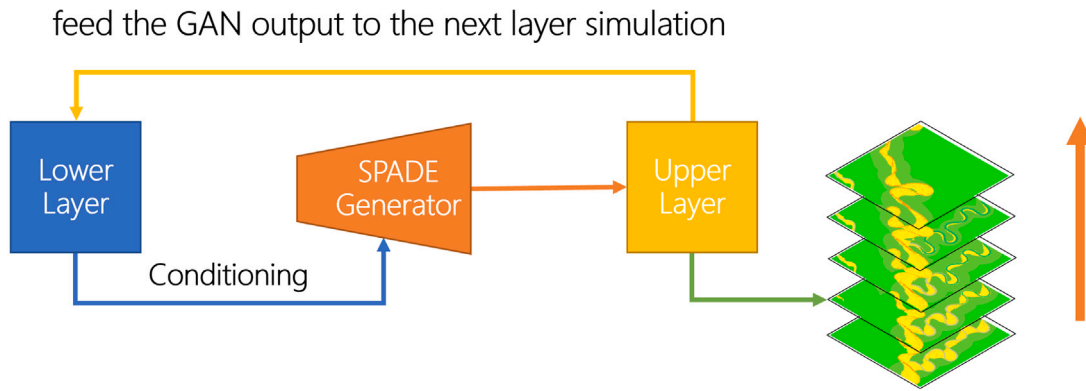


Fig. 2. The workflow of 3D reconstruction process after training. The Lower Layer refers to the conditioning lower-layer data used in the SPADE generator. The Upper Layer is the generated upper layer from the pre-trained SPADE generator.

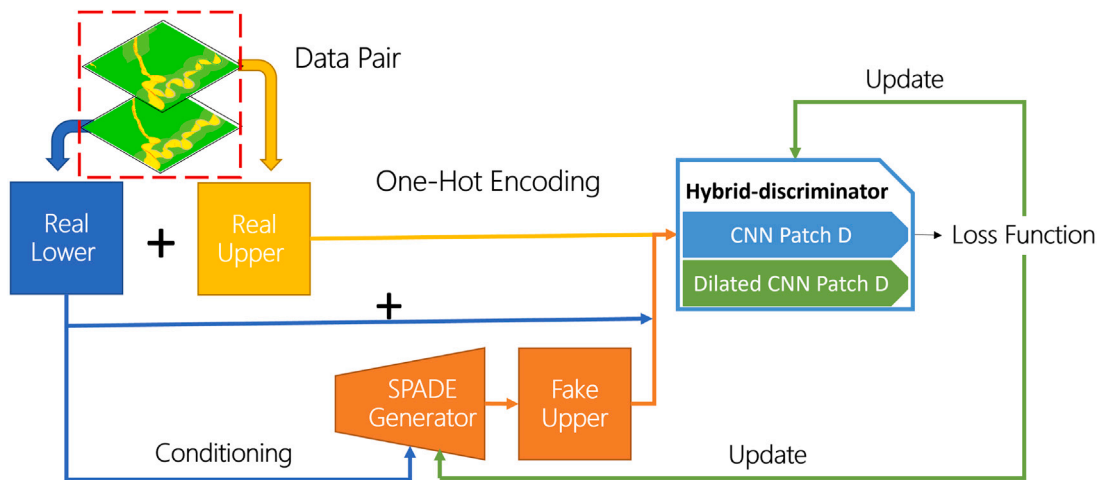


Fig. 3. The framework of training FluvialGAN 3D reconstruction model. Single pair refers to the pair data containing a 2D realisation and the realisation one metre above it. The plus sign denotes vertically concatenating the two 2D realisations before feeding into the discriminator. One-Hot Encoding refers to using the One-Hot Encoder to pre-process data. CNN is short for convolutional neural networks. Patch D refers to the PatchGAN discriminator. The loss function is an improved version of hinge loss by adding the one-centred gradient penalty term. One-Hot Encoder, Hybrid-discriminator and the improved loss function are all from FluvialGAN (Sun et al., 2023).

2.3.1. The training framework of FluvialGAN_3DR

The training framework of FluvialGAN_3DR adopts the same basic configuration as FluvialGAN, except adding a new data pre-processing, replacing the unconditional generator with the SPADE generator, and changing the gradient penalty term in the loss function (see Fig. 3).

For the data pre-processing, FluvialGAN_3DR requires pairs of data, with one training data pair constituting two overlaying 2D layers/slices through the reservoir, instead of a single layer/slice as the discriminator input. The interval between the two layers/slices in each data pair is one metre (the index difference is ten in the training dataset) in this study. The training dataset has 16,000 training images from 25 3D simulations with 640 layers (64 metres thick) representing a range of uncertainty over the variation of avulsion rate (Sun et al., 2022). Therefore, we can have 15,750 data pairs. In each data pair, the *lower layer* is the conditioning data, and the *upper layer* is the target. For simplicity, we use the term ‘*lower layer*’ to denote the conditioning data and the term ‘*upper layer*’ refers to the target at the current reconstruction iteration during training or testing. We use pair data because all 2D realisations from the training dataset can be the input (*lower layer*) and the output (*upper layer*) of the generator during GAN training. Thus, if using a single layer/slice as the training data, the generator only needs to copy the conditioning data (*lower layer*) as its output to get a low loss from the discriminator, resulting in the failure of GAN to predict the target (*upper layer*).

For the model structure, FluvialGAN_3DR uses the same architectures of the generator and the discriminator as FluvialGAN but

replaces the neural network block with the SPADE residual block. We reviewed the SPADE residual block in Section 2.2 whose architecture is in Appendix A Fig. A.1. The detailed configurations of the generator and discriminator are shown in Appendix B.

As for the loss function, FluvialGAN_3DR changes the gradient penalty term in the FluvialGAN loss function (presented in Appendix C) from zero-centred to one-centred based on a trial and error in the workflow configuration. We find the one-centred gradient penalty works well with the two newly proposed training enhancements. The difference between the two gradient penalty terms is the centre value used in the calculation given by Eq. (8) in Appendix C.

2.3.2. Training enhancement 1: Multi-step enhancement

The SPADE generator needs to infer all layers above a given *base layer* but is only trained to predict the layer immediately above the given *base layer* during the GAN training. The accuracy of the predicted layers further above lacks direct supervision from the discriminator. Therefore, we propose to train the GAN model on different layers above the *base layer* by incorporating the reconstruction processes into the GAN training loop, called the multi-step enhancement.

The multi-step enhancement is a nested layer reconstruction loop within the GAN training loop, which introduces two parameters: the *reconstruction step* and the *reconstruction iteration* in each step. At each training iteration, the program runs multiple *reconstruction steps* to train the generator and the discriminator (see Algorithm 1). Each

reconstruction step focuses on a unique layer above the *base layer*, which means different steps target different *upper layers*.

Algorithm 1 Multi-step training enhancement at each training iteration

Input: reference conditioning data and target $[c, x]$, input vector z , the number of reconstruction step N_s , the number of reconstruction iteration at current step N_{si} , the generator $G(\cdot)$, the discriminator $D(\cdot)$, base layer x_0

```

for  $s = 1, \dots, N_s$  do
  for  $i = 1, \dots, N_{si}$  do
    if  $i = 1$  then
       $\tilde{c}_{si} \leftarrow x_0$ ;
       $\tilde{x}_{si} \leftarrow G(z, \tilde{c}_{si})$ ;
    else
       $\tilde{c}_{si} \leftarrow \tilde{x}_{si}$ ;
       $\tilde{x}_{si} \leftarrow G(z, \tilde{c}_{si})$ ;
    end if
  end for
  Train  $G(\cdot)$  on  $[\tilde{c}_{si}, \tilde{x}_{si}]$ 
  Train  $D(\cdot)$  on  $[\tilde{c}_{si}, \tilde{x}_{si}]$  and  $[c_{si}, x_{si}]$ 
end for

```

To get the corresponding target layers, the SPADE generator needs to keep predicting the layer further above iteratively until it reaches the required number of *reconstruction iterations* at the current *reconstruction step*, Fig. 4 shows an example of the program targets on the fifth layer above the *base layer* at current *reconstruction step*. The SPADE generator takes the *base layer* as the initial conditioning data and recycles the *upper layer* as new conditioning data to generate the upper layer until it runs five times the prediction upwards. Then, the program takes the latest *lower* and *upper layers* as the fake data pair and fetches the fourth and fifth layers above the *base layer* from the training dataset as the real data pair. The real and fake data pairs are the input to the discriminator in the training workflow shown in Fig. 3. The rest remains unchanged as the previous workflow shown in Fig. 3.

2.3.3. Training enhancement 2: Conditioning data decay enhancement

Conditioning the target to the layer immediately below it is an assumption of the Markovian process (Gagniu, 2017), but the variation of geomodel layers is not a Markovian process. Each layer should correlate with many *lower layers* with a decaying importance. Thus, we propose accumulating all the previous *lower layers* to predict the target *upper layer*, called the conditioning data decay enhancement.

The conditioning data decay enhancement accumulates layers between the *base layer* and the layer immediately below the target by an exponentially averaging method. In a geomodel, closer layers correlate more strongly when the layer interval is smaller than the channel or the stacked geo-body depth. When two layers are far enough, they should have a very weak correlation and therefore, a forgetting mechanism is necessary for the accumulating method. Inspired by the moment mechanism with a decay rate in gradient-based optimisers (Qian, 1999; Kingma and Ba, 2014), this enhancement accumulates layers with a decaying rate given by Eq. (2),

$$c = \sum_{i=1}^{N_{si}-1} (1-v) \cdot c + v \cdot x_i \quad (2)$$

where c is the conditioning data, initialised as x_0 (*base layer*). x refers to the real data, i denotes the layer index of real data, v is the decay rate, and N_{si} is the number of *reconstruction iterations* at the current *reconstruction step*.

To explore if the decay rate can be a control of tuning the meandering fluvial patterns, we heuristically relate the decay rate to the avulsion rate group number in Sun et al. (2022), called *avulsion rate*

factor, N_{av} . The decay rate in Eq. (2) controls the correlation intensity between layers, which can also be correlated to some geological parameters, e.g. the avulsion rate. In a purely aggrading system, the high avulsion rate results in the meandering river changing its channel quickly across the field. Therefore, one layer is less likely to correlate strongly to the layers further deeper in a high avulsion rate and purely aggrading sedimentary environment. We substitute the decay rate in Eq. (2) with the avulsion rate factor linearly, given by Eq. (3),

$$v = (1 - 0.1) \cdot N_{av} \quad (3)$$

where the avulsion factor N_{av} ranges from 1 to 5; the lower the value, the higher the avulsion rate. After substitution, we get Eq. (4),

$$c = \sum_{i=1}^{N_{iter}-1} 0.1 N_{av} \cdot c + (1 - 0.1) \cdot N_{av} \cdot x_i \quad (4)$$

The conditioning data in the multi-step training enhancement (Algorithm 1) changes to the exponentially averaged generation history after applying the conditioning data decay enhancement. Algorithm 2 shows the pseudo-code of conditioning data decay calculation in one *reconstruction step* of the multi-step training enhancement.

Algorithm 2 Conditioning data decay calculation in one reconstruction step of the multi-step training enhancement

Input: input vector z , the number of reconstruction iteration at current reconstruction step N_{si} , avulsion rate factor N_{av} , the generator $G(\cdot)$, base layer x_0

```

for  $i = 1, \dots, N_{si}$  do
  if  $i = 1$  then
     $\tilde{c}_{si} \leftarrow x_0$ ;
     $\tilde{x}_{si} \leftarrow G(z, \tilde{c}_{si})$ ;
  else
     $\tilde{c}_{si} \leftarrow 0.1 N_{av} \cdot \tilde{c}_{si} + (1 - 0.1) \cdot N_{av} \cdot \tilde{x}_{si}$ ;
     $\tilde{x}_{si} \leftarrow G(z, \tilde{c}_{si})$ 
  end if
end for

```

Output: synthesis data \tilde{x}_{si} , accumulated synthesis condition \tilde{c}_{si}

2.3.4. The implementation of the standard FluvialGAN_3DR

After a trial and error on the associated hyper-parameters, we achieved a stable version of the FluvialGAN_3DR training. The training workflow uses both multi-step and conditioning data decay enhancements. We train the FluvialGAN_3DR model for 50 epochs using ADAM (Kingma and Ba, 2014), whose learning rate is 0.0002 and betas are 0 and 0.9, respectively. The hyper-parameters related to the two enhancements are below:

- (1) For the multi-step enhancement, the number of *reconstruction steps*, N_s , is four, and the number of *reconstruction iterations*, N_{si} , at the four *reconstruction steps* are one, three, five and seven, respectively.
- (2) For the conditioning data decay enhancement, we use either constant or varied decay rates, v , during training. When using varied decay rates, every training data gets a label based on their avulsion rate, called the avulsion rate factor N_{av} , which is used to calculate the decay rate (see Eq. (3)). When using a constant decay rate, we set a constant avulsion rate factor, N_{av} , for every training data.

Two details in this implementation need to be highlighted: the decreased training data and the re-initialisation of the input latent vector during GAN training.

This implementation decreases the number of data pairs to 14,250 in the training dataset. This is because the maximum *reconstruction*

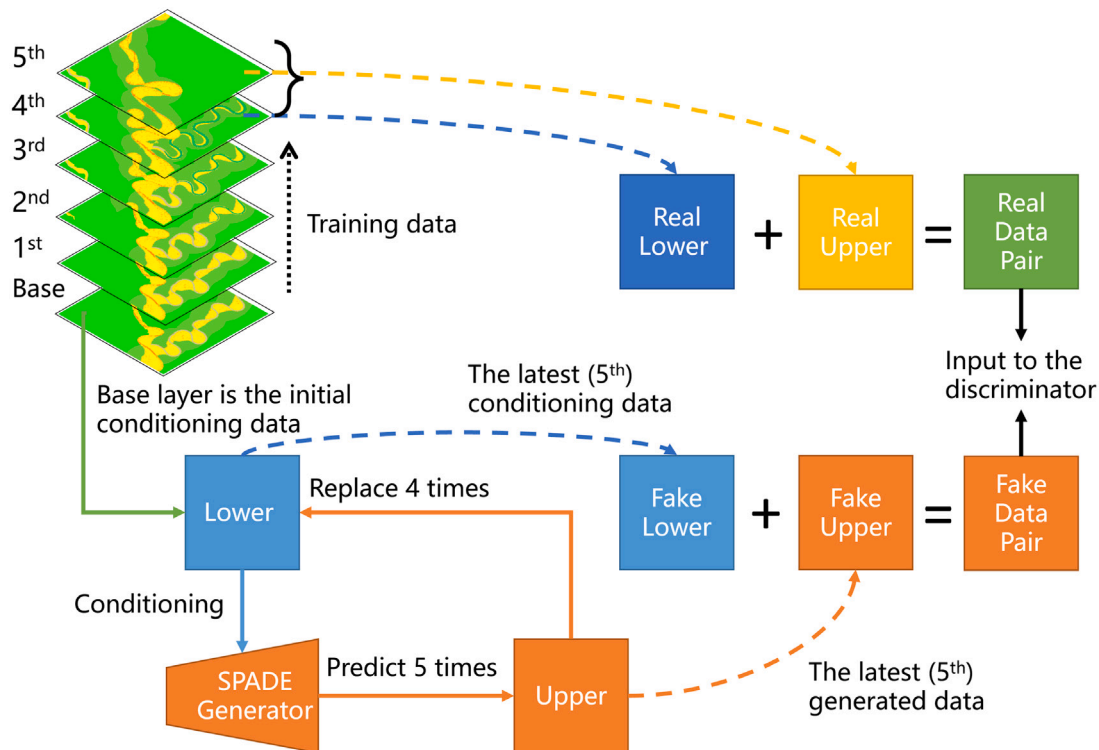


Fig. 4. A schematic diagram of one reconstruction step calculation in the multi-step training enhancement. At this reconstruction step, the generator aims to predict the fifth layer above the given base layer. The real lower and real upper refer to the lower and upper layers fetched from the training dataset, which are the reference data (conditioning data and the target) for the current reconstruction. The training data's base layer is the conditional GAN's initial conditioning data. The lower refers to the conditioning data of the conditional (SPADE) generator at each reconstruction iteration. The upper refers to the generated data at each reconstruction iteration.

iteration in multi-step enhancement is seven. Therefore, the training program requires seven layers further above the *base layer*, reducing useable training data as the *base layer* in the training dataset.

At each training iteration, the training program reinitialises a random vector as the input vector, z , for the generator and fetches the avulsion rate factors used in the reconstruction. The input vector remains unchanged at the same *reconstruction step* for simplicity. We know that the input vector can work as a 'seed' in the stochastic simulation, leading to different results. However, the discussion about the effect of varying the input vector z in reconstruction is beyond the scope of this paper. The studies below all use the same input vector in both Fluvial GAN and 3D reconstruction to simulate a single 3D model.

2.4. Quantitative measure of 3D facies model

Sand-body connectivity and facies proportions are two fundamental characteristics of 3D facies models that impact their dynamic performance. Here we consider the sand proportion as the volume ratio between sand-prone facies and the total field, which counts the number of cells taken by channel lag, point bar and sand plug and divided by the total number of cells in practice. The sand connectivity refers to the largest connected body in the sand-prone facies definition (King, 1990). We use a Python package, scikit-image, to pick the largest connected body (van der Walt et al., 2014).

Previous studies described reservoir connectivity using an S-curve that fits the relationship between the sand proportion (NTG) and the connectivity based on percolation theory (King, 1990; Larue and Hovadik, 2006). The S-curve plot shows a rapid increase of connectivity once the proportion reaches a *percolation threshold* (see Fig. 5 a), which is about 0.3 for the 3D models (King, 1990; Larue and Hovadik, 2006). However, real geological systems may diverge from the theoretical percolation threshold at 0.3 because the reservoir's geological features can increase or decrease connectivity for a given NTG. For instance,

more sinuous sand-filled channels will result in greater overall connectivity than straight channels, which would, on average, require a larger proportion to achieve the same connected volume (Larue and Hovadik, 2006). For an ensemble of geology realisations, we would expect individual realisations to scatter around the S-Curve (in the cascade zone), where the average response of the ensemble is shifted to the left or right, where the geology increases or decreases the connectivity. This allows individual models to have significantly different connectivity/NTG relationships than predicted by percolation alone.

We plot the sand connectivity and proportion of fifty 3D facies models with 32 metres of thickness as the reference. The original twenty-five 3D FLUMY simulations in GAN River-I are 64 m thick. We split them in the middle to create this fifty-point reference sand connectivity against the sand proportion plot (see Fig. 5 b). As GAN River-I is a low NTG meandering dataset (Sun et al., 2022), those models' sand proportions are mostly about 0.2, smaller than the 3D percolation threshold value (0.3). As we see in Fig. 5 b), the scatter of our FLUMY data set diverges significantly from the idealised S-Curve and achieves higher connectivities at a lower percolation threshold at *high avulsion* (and therefore lower sinuosity). This is due to the combination of channel migration and meandering mechanisms. The high scatter in connectivity for a given NTG suggests we expect high uncertainty in the predicted connectivity of models of this type of geology which our GAN should honour in its realisations.

3. Results

We first follow the ablation study manner to demonstrate the effect of our training enhancements on this conditional GAN-based training framework. We create three cases with the same settings to explore the impact of the chosen training enhancements — multi-step and conditioning data decay (see Table 2). After training, we set up two methods of initiating the *base layer* for 3D reconstruction to test the

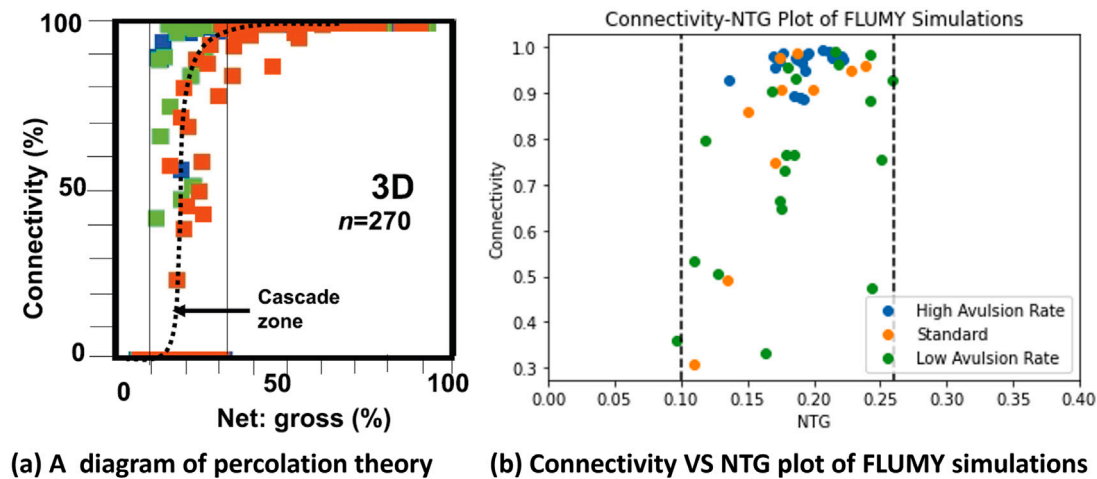


Fig. 5. Plots of sand connectivity in 3D against proportion. (a) A schematic diagram of percolation theory, showing the S-curve and Cascade zone. Modified after (Larue and Hovadik, 2006). (b) A plot of sand connectivity against proportion for 50 FLUMY simulations with 32 metres of thickness. The area within the two black dashed lines is the Cascade zone.

Table 2

Cases in the ablation study.

GAN Cases	Multi-step enhancement	Conditioning data decay enhancement	N_{av} during training
Case 1	×	×	NA
Case 2	✓	×	NA
(Standard) FluvialGAN_3DR	✓	✓	[1,5] depends on the training data label
High avulsion rate variant	✓	✓	1
Low avulsion rate variant	✓	✓	5

✓denotes Yes. × denotes No. NA denotes not applicable. N_{av} denotes the avulsion rate factor.

performance, catering to the demands of different difficulties. One uses a given slice from the GAN River-I dataset as the *base layer* to reconstruct 3D models. The other uses Fluvial GAN to randomly simulate 100 realisations to evaluate the 3D reconstruction's performance as the test case.

Then, we further explore the impact of the avulsion rate factor, N_{av} , on the FluvialGAN_3DR by decoupling this factor from the avulsion rate label in training data. Compared to the standard configuration of FluvialGAN_3DR, we create two FluvialGAN_3DR variants with two extreme constant values of N_{av} during training, named the high avulsion rate variant and the low avulsion rate variant (see Table 2). We train the two variants using the same GAN model structure and training method settings as the standard FluvialGAN_3DR. After training, we conduct two experiments by feeding the same sets of input vectors to the three configurations (one standard and two variants), which guarantees they have the same *base layer* set. The first experiment compares the two variants in building 3D samples using the same *base layer* set to study whether a constant N_{av} configuration works. The second study tests the three configurations of pre-trained FluvialGAN_3DR by varying the N_{av} value during the reconstruction process (testing stage) to explore the impact of N_{av} on the reconstructed pattern.

We analyse the results in both quantitative and qualitative ways, as quantitative scores are less subjective, while qualitative analysis often delivers information on high-level complexities that is hard to quantify.

3.1. Ablation study 1: The effect of the multi-step enhancement

We pick the same slices from the training dataset as the *base layer* to compare the performance of Case 1 and Case 2. Both Case 1 and Case 2 initially predict plausible *upper layers*. However, Case 1 gradually loses realism as the reconstruction program builds realisations upwards (see Fig. 6 a). The reconstructed realisations only show a big chunk of point bars mingled with some broken channel fills. In contrast, Case 2 can still create plausible meandering channels and place point bars at the

inner banks, though the overall visual quality is not as good as FLUMY simulations (see Fig. 6 b).

This ablation study demonstrates the conditional GAN trained with the multi-step enhancement significantly outperforms the one without it. Visual inspection of the data shows the *upper layers* (above the 10th) from Case 2 are more plausible than those from Case 1, with no need for quantitative scores to prove this point (please refer to our GitHub repository¹ for additional results of Case 1). To build 3D models with a flexible thickness (number of layers), preserving geological realism is necessary during reconstruction, which warrants using a multi-step process.

3.2. Ablation study 2: The effect of the conditioning data decay enhancement

In this study, we quantitatively compare the reconstructions from Case 2 (without conditioning data decay) and FluvialGAN_3DR by building 100 3D simulations with 32 metres of thickness (1 *base layer* and 31 reconstructed *upper layers*). We use a pre-trained Fluvial GAN to create the test set of *base layers* and connect Fluvial GAN and the reconstruction model to complete the 3D simulator, Fluvial GAN 3D. We use the same input vectors for both 3D simulators based on Case 2 and FluvialGAN_3DR to avoid the bias of different initialisation. As the avulsion rate factor is evenly distributed during training FluvialGAN_3DR, we assign its value from 1 to 5 for every 20 samples when using FluvialGAN_3DR to reconstruct the test set. We use the sand proportion and connectivity as two indicators to plot GAN simulations versus reference models from GAN River-I.

Facies models from FluvialGAN_3DR show wider coverage of sand proportion and connectivity than those from Case 2, better matching the reference ranges. This confirms the conditioning data decay enhancement improves the performance of conditional GAN-based 3D

¹ <https://github.com/GeoDataScienceUQ/FluvialGAN3D/tree/main/Case1>.

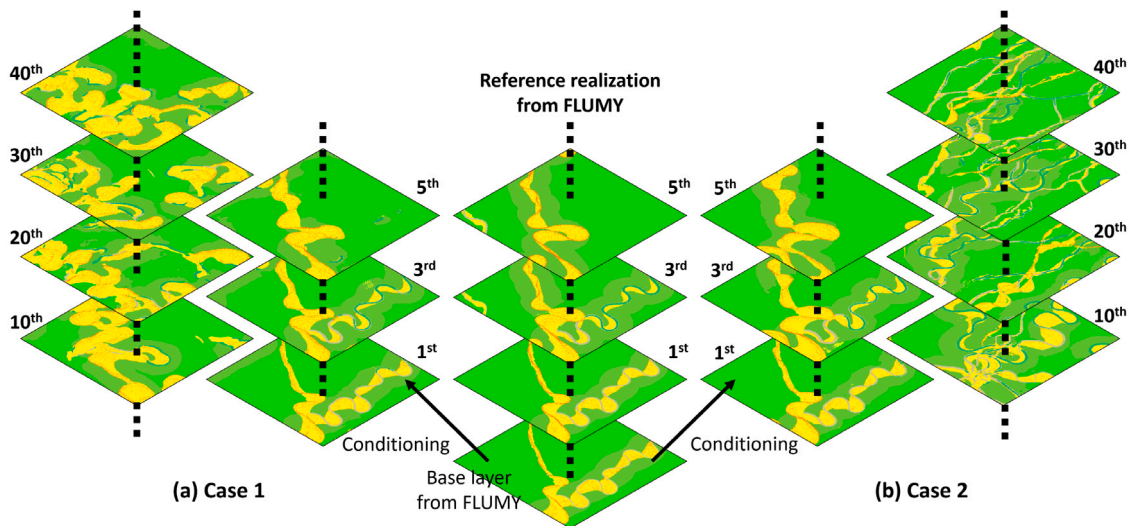


Fig. 6. Comparison of meandering patterns between Case 1 and Case 2. (a) is Case 1, and (b) is Case 2.

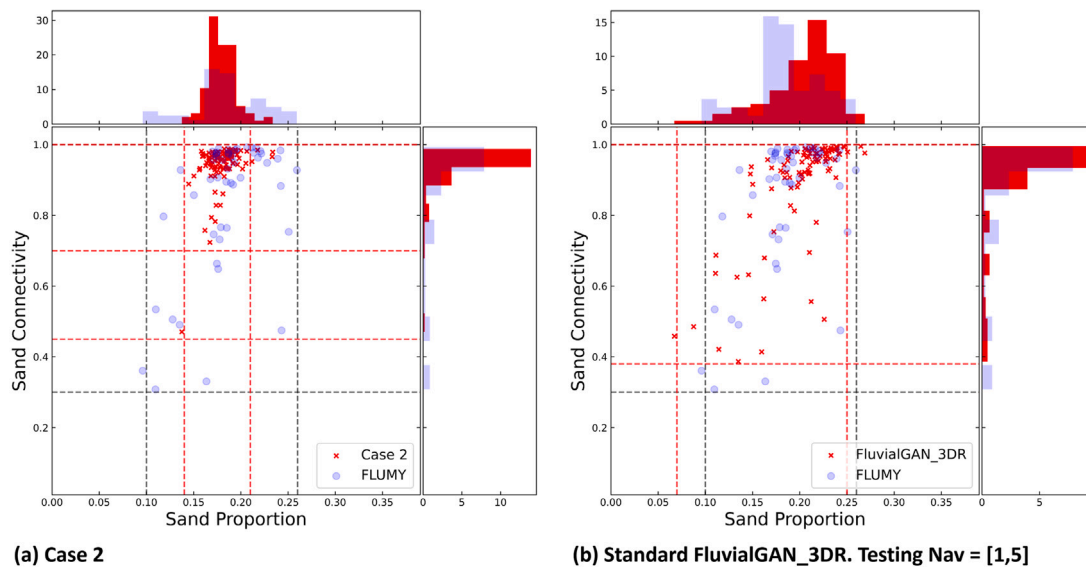


Fig. 7. Comparison of sand connectivity against proportion plots between Case 2 and FluvialGAN_3DR. The N_{av} value of the FluvialGAN_3DR ranges from 1 to 5 during training. (a) is Case 2, and (b) is the standard FluvialGAN_3DR with varying N_{av} during the reconstruction process. The blue dots are the reference samples from FLUMY. The red cross markers are samples from GANs.

reconstruction (see Fig. 7). Samples from FluvialGAN_3DR present both low and high-connectivity models and cover the range represented by GAN River-I. In contrast, Case 2 results are mostly high connectivity models, given the same initialisation. The Cascade zone of the Case 2 plot is considerably smaller, illustrating it fails to honour the connectivity uncertainty represented by FLUMY. Therefore, with the help of the conditioning data decay enhancement, the conditional GAN-based model learns better than those without it. We discuss the impact of the avulsion rate factor in the following two sections, Sections 3.3 and 3.4.

We further investigate what features of fluvial geology Case 2 did not capture and analyse how this impacts the resulting connectivity range. The connectivity plot based on different avulsion rates reveals that the low connectivity models are more likely to appear in the low avulsion rate settings (see Fig. 5b). Considering the sand proportion is similar, low avulsion rates lead to the channel having more time to develop laterally and sand extending wider at the inner bank. After avulsion, sand accretes at other places in a distance, making the sand bodies less likely to be connected. In comparison, high avulsion rate

settings cause the river to change its channel frequently and place the elongated sand bodies widely in the field, increasing the chance of sandbodies connecting to each other.

We visually checked all 2D slices from bottom to top and found nearly all *upper layers* from Case 2 feature less-developed channel meanders with elongated channel shapes, which is consistent with the relationship between connectivity and avulsion rate setting in GAN River-I. The vertical cross-section of 3D models from Case 2 proves that the conditional GAN changes to a high avulsion rate no matter what kind of realisation as the *base layer* (see Fig. 8). Fig. 8 shows an example of Case 2 shifting from sheet-type to ribbon-type sandbodies, which is inconsistent with the pre-set setting. The conditional GAN in Case 2 tends to abandon and reinitialise the channel instead of developing it. We will further discuss this in Section 4.1.

In contrast, FluvialGAN_3DR can produce well-developed channel meanders with more sinuous channel shapes and lateral-extended sandbodies. Given the same example shown in Fig. 8, FluvialGAN_3DR develops sinuous channels instead of frequently abandoning the old channels, favouring the sheet-type sandbodies (see Fig. 9). We set N_{av}

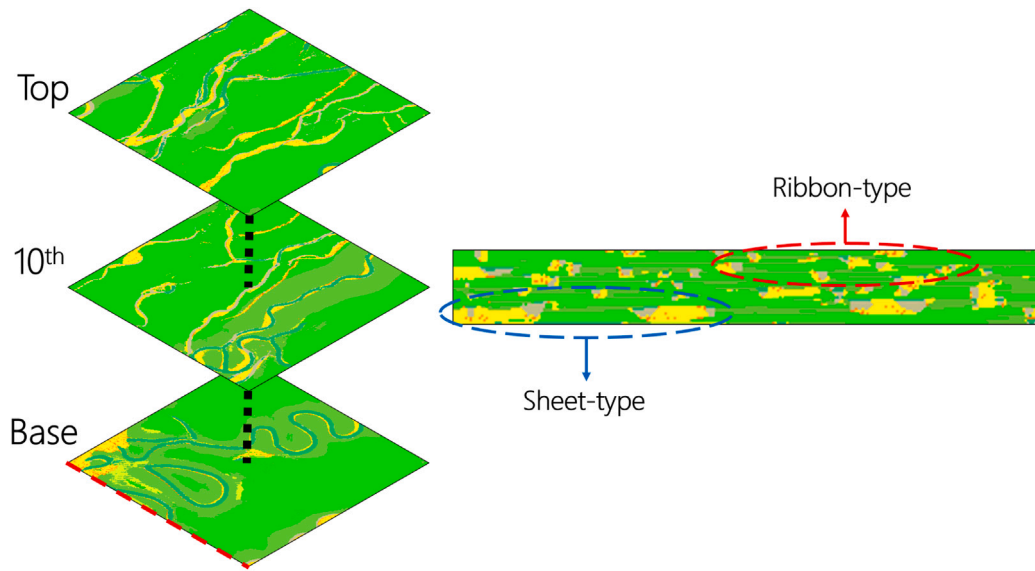


Fig. 8. Horizontal and vertical slices of an example initialised with a developed meandering pattern from Case 2.

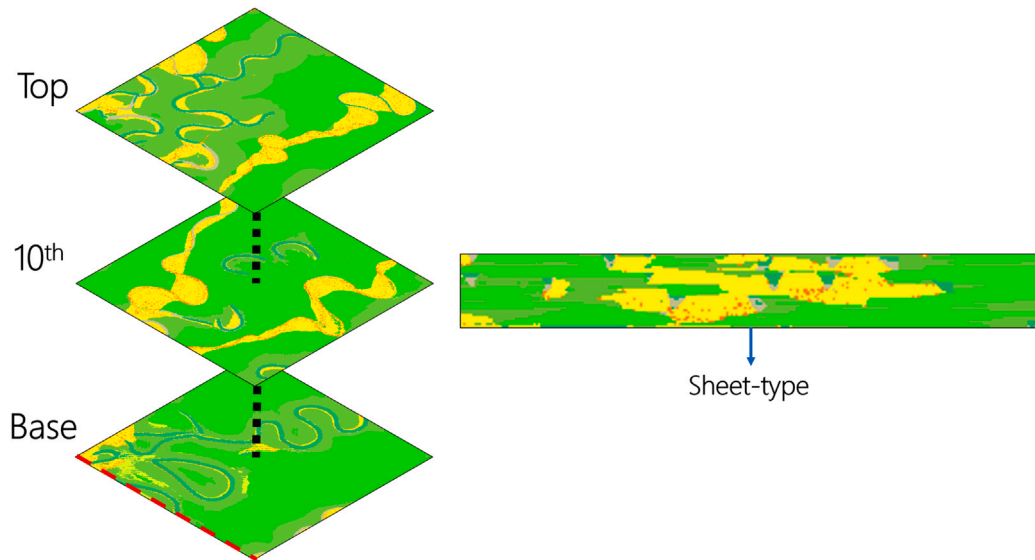


Fig. 9. Horizontal and vertical slices of an example initialised with a developed meandering pattern from FluvialGAN_3DR.

to 5 when we reconstruct this sample and will present the impact of N_{av} on FluvialGAN_3DR in Section 3.4.

3.3. Variant study 1: The effect of avulsion rate factor during training

We compare the reconstructed 3D facies models of the two variants and the standard FluvialGAN_3DR by giving the same base layer set. Then, we calculate the sand proportion and connectivity of those 3D facies models (see Fig. 10). Compared to the standard FluvialGAN_3DR, the high avulsion rate variant has a narrower Cascade zone in the connectivity plot, while the low avulsion rate variant seems to achieve a similar spread of data points.

We visually analyse the reconstructions from the high and low avulsion rate variants to explore the reason behind the different connectivity behaviours. Similar to Case 2, the high avulsion rate variant also tends to avulsion instead of developing channels (see Fig. 11a). In contrast, the low avulsion rate variant develops the type of meandering models given to it (see Fig. 11b). This difference reflects the relationship between N_{av} and spatial correlation strength. According to Eq. (4)

and Algorithm 2, a smaller value of N_{av} gives the biggest weight to the layer immediately below, with the weight decreasing for deeper layers. Therefore, the latest generation (the layer immediately below the current target) significantly impacts the high avulsion rate variant, while earlier generations (deeper layers) decay rapidly and have a very limited influence on the high avulsion rate variant.

3.4. Variant study 2: The effect of varying avulsion rate factor during reconstruction

We further investigate the impact of varying avulsion rate factor, N_{av} , values during the reconstruction process (testing stage) to explore if it is a potential way of controlling GAN's reconstruction to reflect different avulsion rate settings. We use the standard and variant configurations of FluvialGAN_3DR to reconstruct the same test sets with the two extreme N_{av} constants, one and five, during the reconstruction process. Then, we calculate each case's sand proportion and connectivity and visualise all slices for every sample to compare the results quantitatively and qualitatively.

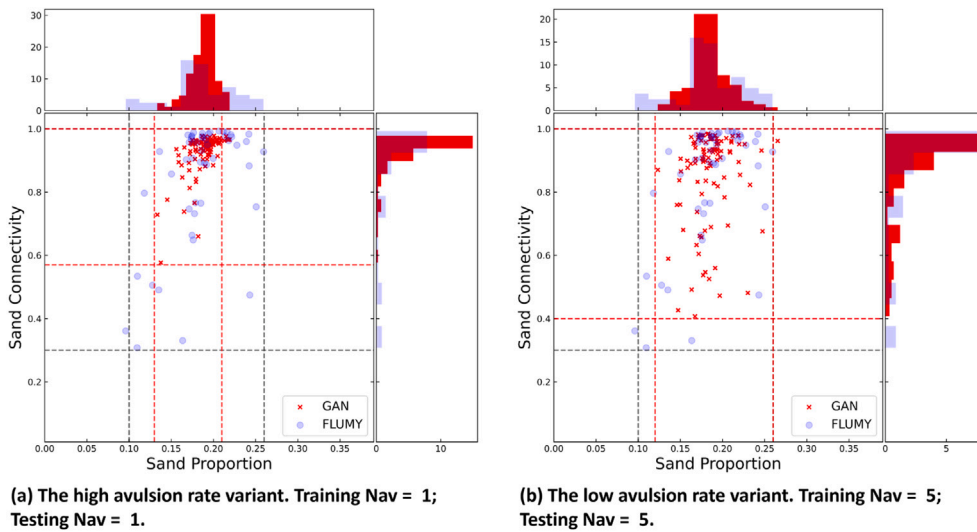


Fig. 10. Comparison of sand connectivity against proportion plots between the high and low avulsion rate variants. (a) is the high avulsion rate variant. (b) is the low avulsion rate variant. The blue dots are the reference samples from FLUMY. The red cross markers are samples from GANs.

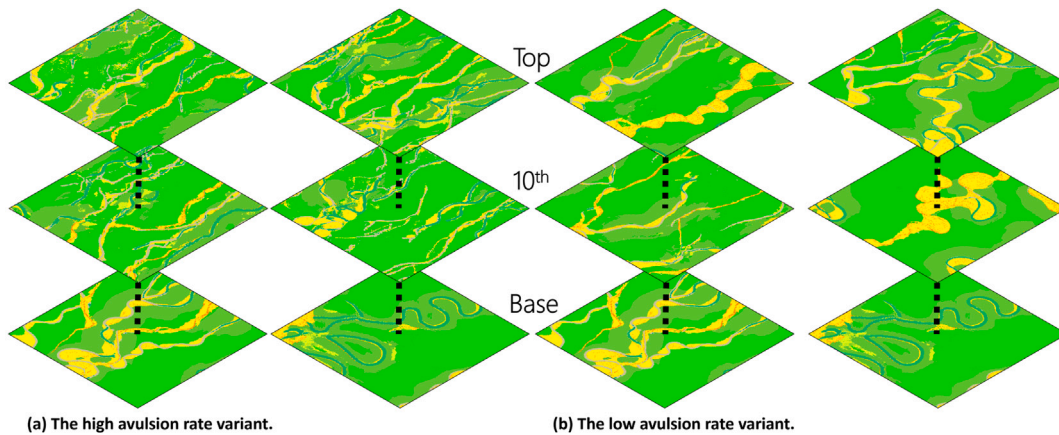


Fig. 11. Comparison of meandering patterns between the high and low avulsion rate variants. (a) is the high avulsion rate variant ($N_{av} = 1$ during training). (b) is the low avulsion rate variant ($N_{av} = 5$ during training).

The high avulsion rate variant performs worse if we change its N_{av} to another end value in the reconstruction processes. When we set N_{av} to 5, the high avulsion rate variant ($N_{av} = 1$ during training) tends to generate more background (overbank facies) after around 10–20 reconstruction iterations, which causes the proportion-connectivity plot shift to the left side (see Fig. 12).

The low avulsion rate variant can reflect the change of N_{av} values in the reconstruction to some level but performs less ideally in some cases. The low avulsion rate variant ($N_{av} = 5$ during training) can still develop the meandering channels instead of giving blank layers when we set its N_{av} to 1 during testing, while the proportion-connectivity plot reasonably narrows to a smaller range (see Fig. 13). However, given a less developed meandering realisation that can be abandoned later or further developed, the low avulsion rate variant tends to avulsion no matter how we change the N_{av} values during testing reconstruction.

The standard FluvialGAN_3DR shows a clear connectivity shift when we change the N_{av} from one to five (see Fig. 14). We pick the same less developed meandering realisation and simulate upwards using FluvialGAN_3DR with different N_{av} . When N_{av} equals one during reconstruction, the standard FluvialGAN_3DR produces elongated channels (see Fig. 15a). When we change N_{av} to five, the standard FluvialGAN_3DR creates a more meandering channel, reflecting the different geological settings (see Fig. 15b).

4. Discussion

4.1. The generation history makes FluvialGAN_3DR more stable and balanced

Cases 1 and 2 do not incorporate the generation history properly into the new prediction. Case 1 completely ignores the generation history's impact and predicts a new *upper layer* based only on the latest generation. This behaviour of forgetting history is the same as the Markovian process that assumes X_{t+1} depends on X_t (Gagniuc, 2017). Case 2 relies on neural networks (the discriminator) to penalise the predicted *upper layers* further above the *base layer*, assuming the generation history is involved in the sequential reconstruction throughout the neural networks (conditional generator). However, we do not train this conditional GAN in a recurrent way, which connects every new generation directly to the input of the next generation. Instead, we disconnect the generator output before feeding it to the next generation's input end, detaching the gradients from the computation. This choice is due to the limitation of the GPU memory. Therefore, Case 2 also does not explicitly include the generation history in the next prediction, as its conditioning data is still the latest generation (the layer directly below the current target).

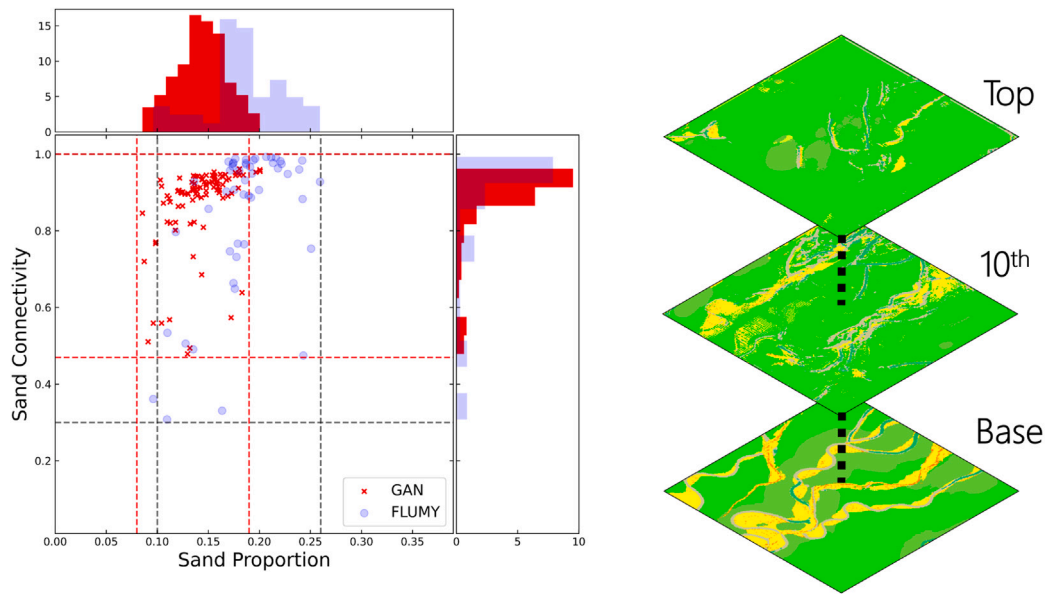


Fig. 12. Sand connectivity against proportion plot and slices visualisation of the high avulsion rate variant ($N_{av} = 1$ during training) when setting N_{av} to 5 in the reconstruction process.

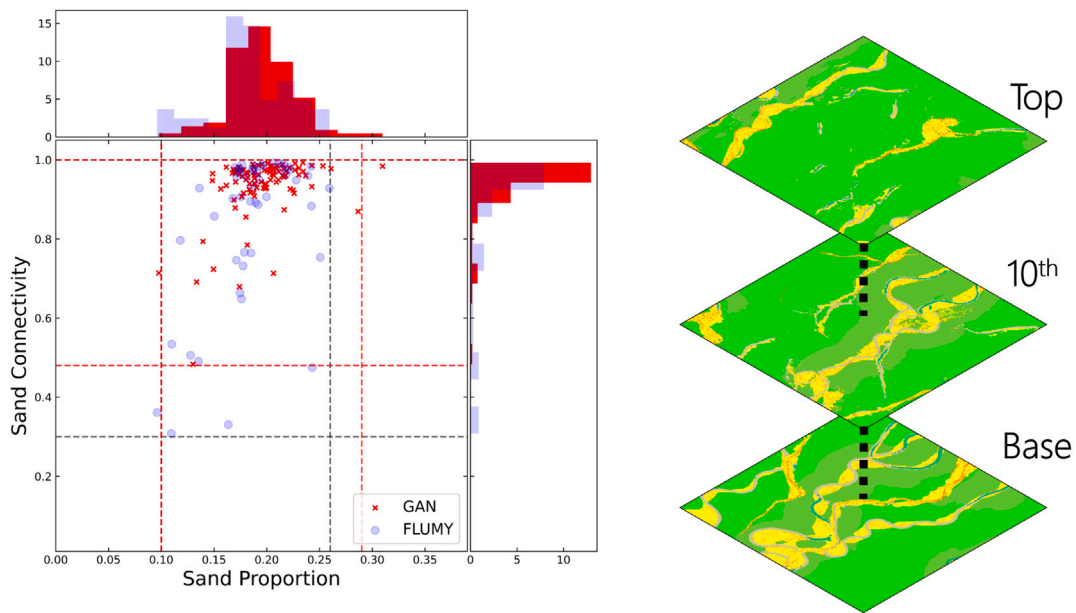


Fig. 13. Sand connectivity against proportion plot and slices visualisation of the low avulsion rate variant ($N_{av} = 5$ during training) when setting N_{av} to 1 in the reconstruction process.

Case 2 works fine for high avulsion rate settings but not for low avulsion rates according to the results in Section 3.2. A layer of a 3D model with low avulsion rates has stronger spatial correlations to nearby layers because the channel is less likely to switch positions as lower avulsion rates increase the probability that the new layer will have a channel located close to the previous layer. The indirect correlation between the prediction and its generation history in Case 2 causes the information from layers further below the target to have a very weak impact on the prediction. This lack of spatial correlation accounts for Case 2 missing the well-developed meandering fluvial pattern at its *upper layers*.

On the other hand, FluvialGAN_3DR direct introduces generation history into the conditioning data with a decaying mechanism to honour the correlations between layers, as closer layers have stronger

correlations. This method follows the same idea of the autoregressive model that predicts the output value based on its own historical values (Box et al., 2015). The difference is the classical/deep autoregressive models generate a single data/image element-by-element/pixel-by-pixel (Box et al., 2015; Van den Oord et al., 2016; Menick and Kalchbrenner, 2018; Hoogeboom et al., 2021). By contrast, our method directly generates the facies model/image by incorporating historical generations/images into the conditioning data. So, our method can be regarded as a conditional GAN trained in an autoregressive way to create a 3D cube layer-by-layer.

The FluvialGAN_3DR framework opens space for investigating other methods of incorporating generation history and building correlations with geological parameters. The conditioning data decay method uses a constant decay rate to bias the weights of previous *lower layers*. One can also use more advanced methods to incorporate generation history,

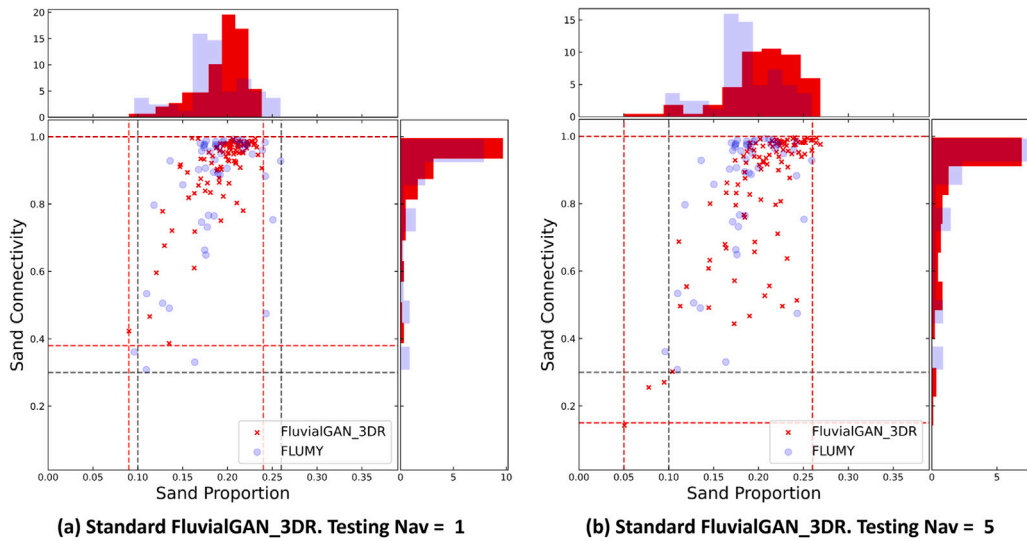


Fig. 14. Comparison of sand connectivity against proportion plots when changing N_{av} from 1 to 5 in FluvialGAN_3DR reconstruction process. (a) is setting N_{av} to 1 during reconstruction. (b) is setting N_{av} to 5 during reconstruction. The blue dots are the reference samples from FLUMY. The red cross markers are samples from GANs.

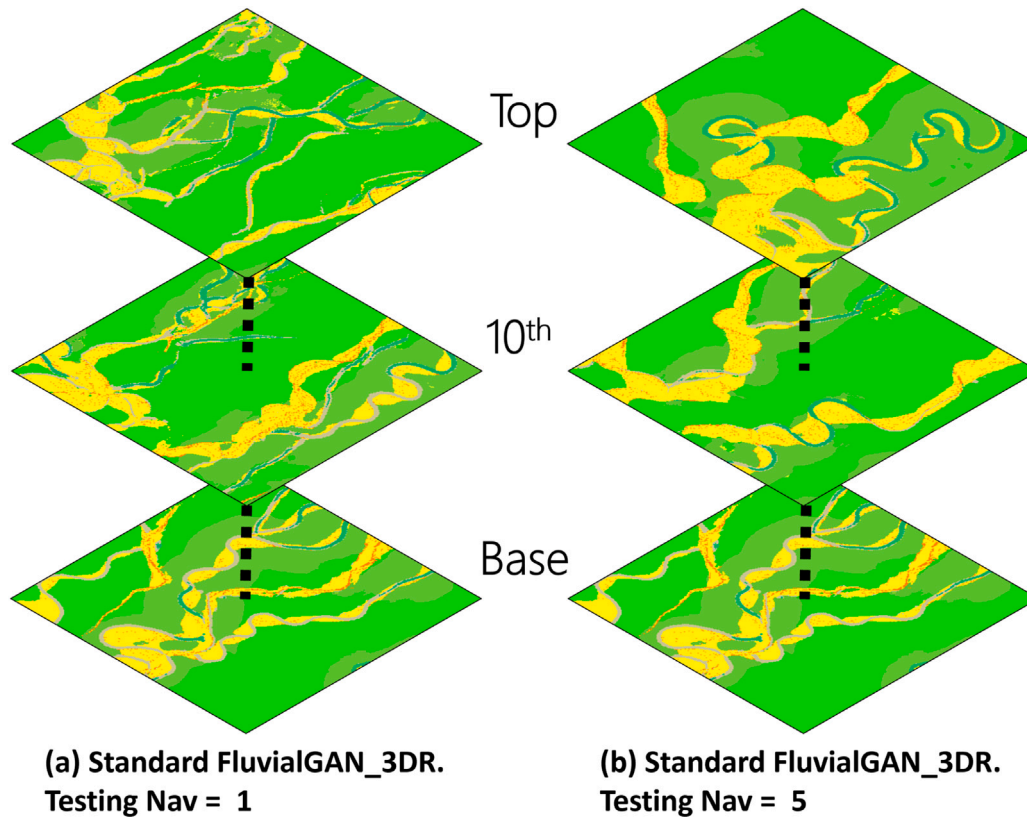


Fig. 15. Comparison of meandering patterns when changing N_{av} from 1 to 5 in the standard FluvialGAN_3DR reconstruction process. The N_{av} value of the standard FluvialGAN_3DR ranges from 1 to 5 during training. (a) is setting N_{av} to 1 during reconstruction. (b) is setting N_{av} to 5 during reconstruction.

e.g. recurrent network, long short-term memory (LSTM) units, CNN-LSTM architectures, etc. (Rumelhart et al., 1985; Williams and Zipser, 1989; Hochreiter and Schmidhuber, 1997; Donahue et al., 2015). The increase in GPU demand is foreseeable, and potential robustness issues associated with those methods may occur, which is worth further investigation. We use the group label, the avulsion rate factor N_{av} , to rank the training data qualitatively. Research on building quantitative relationships between the geological parameter and the layer correlation should make the control of geological patterns more meaningful and convincing.

4.2. Advantages and remaining challenges of fluvial GAN 3D modelling

The first advantage of Fluvial GAN 3D modelling is it allows the user to define the reservoir thickness or the number of layers because it is an iterative process. This flexible thickness feature makes the pre-trained GAN models more reusable for building up 3D models, avoiding wasting time retraining them when needed to apply them to a similar field with different thicknesses. Also, this iterative process has a lower computational cost as it deals with 2D data and detaches the gradients to reduce the GPU memory demand.

The second advantage is the Fluvial GAN 3D modelling provides an explicit parameter, the avulsion rate factor, to reflect the global setting. The classical conditional GAN model takes global features as an extra input to the neural networks, impeding the user from understanding how it impacts the generation result. In contrast, our method blends our understanding of the vertical correlation between layers in a purely aggrading system into the conditional GAN training and generating processes, making the global features' impact on GAN generation more interpretable.

Also, the iterative approach of the FluvialGAN_3DR mimics a purely aggrading depositional process, making it possible to be applied to other sedimentary settings of a purely aggrading system. Though we only verify this approach using a low NTG meandering fluvial dataset, a modeller could also try this iterative approach for modelling a high NTG case or other sedimentary reservoirs by training on corresponding datasets if they are purely aggrading. This is because this approach learns to generate the *upper layer* layer by layer based on its generation history, which is consistent with the aggradation of other sedimentary reservoirs.

The major limitation is the conditioning data decay enhancement cannot reflect the case of incisions. First, the training dataset used in this study simulates 3D realisations in a purely aggrading system, and no incision is included. Therefore, we cannot verify the Fluvial GAN 3D modelling in an incision case. Second, the assumption in the conditioning data decay enhancement correlates the avulsion rates and the correlation intensity between layers/vertical distance is valid in a purely aggrading system, but not for the case of incisions. This is because the facies distribution vertically is no longer completely correlated with time if the incisions are present.

Another limitation is that Fluvial GAN 3D modelling cannot fully customise the 3D model's spatial sizes. This is because the generator structure in the standard configuration is not fully convolutional neural network-based. Therefore, the standard Fluvial GAN 3D cannot change the generator's output size, which is the lateral size of the 3D simulations from the Fluvial GAN 3D modelling. To the best of our knowledge, current methods of enlarging GAN generation size still rely on increasing some layers' size (including the latent vector size) of a pre-trained model to proportionally enlarge the generations' size (Laloy et al., 2018; Song et al., 2022). How to freely define GAN generations' size remains the problem to be solved.

Data conditioning is not yet applied to the Fluvial GAN 3D modelling. Though we use the conditional GAN technique, this study does not focus on the data conditioning as such, though the proposed approach can be applied to seismic conditioning. Condition the GAN generation to observed point data has attracted much attention and yielded several publications suggesting data conditioning workflows in different ways (Chan and Elsheikh, 2019; Zhang et al., 2019; Song et al., 2022). We are keen to combine our model and those methods in future work. Our approach processes 2D data iteratively and, therefore, is flexible to incorporate the conditioning techniques applied to 2D images. For example, conditional normalisations, e.g. SPADE, can take different types of conditioning data as an extra input of conditional GAN modelling. Particularly, Park et al. (2019) trained their SPADE generator using different datasets with more semantic classes (e.g. 35, 150, 182) than our study (7 indicator maps). Therefore, adding more maps, e.g. acoustic impedance or amplitude maps to the input of SPADE as the conditioning data, is a potential way of conditioning Fluvial GAN 3D generations.

5. Conclusions

We developed a new 3D reconstruction framework, named FluvialGAN_3DR, based on conditional GAN, which can simulate 3D reservoir facies models with flexible thickness by mimicking a purely aggrading depositional process upwards. FluvialGAN_3DR successfully reconstructs a customised number of layers, preserving the meandering

channel geometry and placement of associated facies with the help of the multi-step training enhancement. We introduce historical GAN generations into the conditioning data in a decaying manner, imitating vertical correlations among geomodel layers in a purely aggrading system. This conditioning data decay enhancement prevents FluvialGAN_3DR from collapsing to generate high avulsion rate models, enriching the model diversity and converging a good range of sand connectivity.

We presented that the avulsion rate factor in the conditioning data decay enhancement provides an alternative way of controlling GANs' generation based on sedimentary settings. This avulsion rate factor directly adjusts the conditioning data instead of working as extra conditioning data, making the users understand the consequence of tuning this factor more easily.

This work is subject to assumptions inherent to the training dataset and, therefore, has several remaining challenges. We simulate the process-based models using a purely aggrading setting and get the training data by slicing horizontal layers on the preserved sedimentary facies models. Thus, some assumptions of our method are invalid in some cases. For example, the conditioning data decay enhancement is invalid in the presence of an incision. This needs a relevant dataset to be appropriately prepared and a redefined output end of the generator. Some methods proposed by other researchers, e.g. data conditioning and using full CNN design to allow changing the generator output size, have not been applied to our proposed approach, which can be further enhancements.

CRedit authorship contribution statement

Chao Sun: Conceptualization, Methodology, Software, Validation, Data curation, Writing – original draft, Visualization. **Vasily Demyanov:** Writing – review & editing, Supervision. **Daniel Arnold:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The dataset used in this paper has been published in data in brief and cited in the manuscript

Acknowledgements

The work contained in this paper contains work conducted during a PhD study undertaken as part of the Natural Environment Research Council (NERC) Centre for Doctoral Training (CDT) in Oil & Gas [grant number NEM00578X/1]. This project is sponsored by Heriot-Watt University via their James Watt Scholarship Scheme whose support is gratefully acknowledged. The authors would like to acknowledge MINES ParisTech for their software *FLUMYTM*, Process-based channelised reservoir models. Copyright © MINES PARIS-PSL/ARMINES. Free download from <http://cg.ensmp.fr/flumy>. We would like to thank Prof. Ahmed ElSheikh for his kind discussion and suggestions and thank Dr. Guillaume Rongier and the anonymous reviewer for their valuable feedback.

Code availability section

FluvialGAN3D
Contact: cs157@hw.ac.uk
Program language: Python
The raw data in this paper are available for downloading at the link: <https://github.com/GeoDataScienceUQ/FluvialGAN3D>

Appendix A. The schematic diagram of the SPADE generator

See Figs. A.1 and A.2.

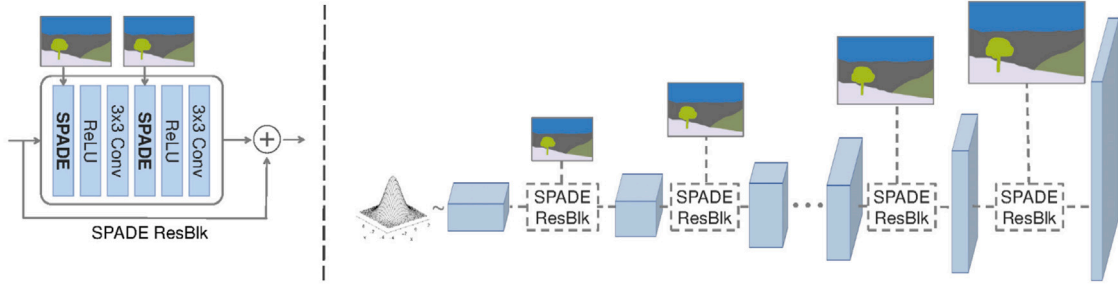


Fig. A.1. A schematic diagram of SPADE generator. The SPADE ResBlk refers to a block of neural networks in a residual block architecture, shown on the left side, featuring a skip connection from its input to output. 3×3 Conv refers to the convolutional layer with 3×3 kernels. ReLU refers to the rectified linear unit, a non-linear activation. SPADE is a conditional normalisation layer whose architecture is shown in Fig. A.2.

Source: From Park et al. (2019).

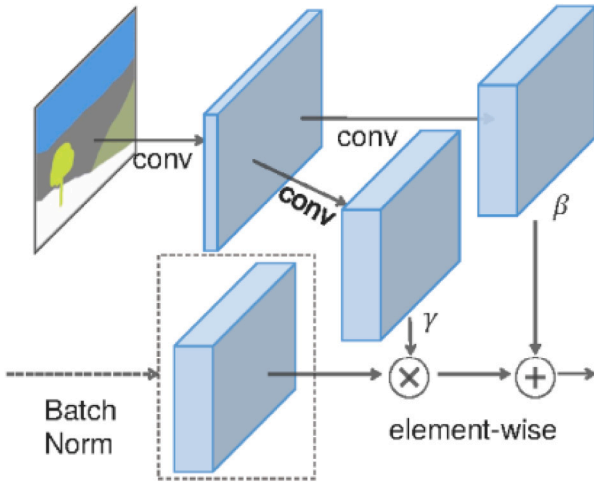


Fig. A.2. A schematic diagram of spatially-adaptive denormalisation (SPADE). *conv* denotes the convolutional layer. Batch Norm is short for the batch normalisation layer but also refers to other unconditional normalisations, such as instance normalisation. γ and β denote the learned scaling and bias parameter in Eq. (1) respectively.

Source: From Park et al. (2019).

Appendix B. FluvialGAN_3DR model architecture

B.1. Generator: SPADE generator

See Table B.1.

Table B.1
SPADE generator architecture.

Layer	State size
Linear (128, 32768)	(Batch size, 32768)
Reshape	(Batch size, 512, 8, 8)
SPADEResBlk, Up sample (2)	(Batch size, 512, 16, 16)
SPADEResBlk, Up sample (2)	(Batch size, 256, 32, 32)
SPADEResBlk, Up sample (2)	(Batch size, 128, 64, 64)
SPADEResBlk, Up sample (2)	(Batch size, 64, 128, 128)
SPADEResBlk, Up sample (2)	(Batch size, 32, 256, 256)
SPADEResBlk, 3X3Conv-7, SoftMax	(Batch size, 7, 256, 256)

3X3Conv-7 denotes 3X3 convolutional layer with seven filters.

B.2. Discriminator: Hybrid-discriminator

See Tables B.2 and B.3.

Table B.2
CNN-based PatchGAN discriminator.

Layer	Conv parameters (kernel size, stride, dilation)
Conv-32, Leaky ReLU	(4, 2, 1)
Conv-64, IN, Leaky ReLU	(4, 2, 1)
Conv-128, IN, Leaky ReLU	(4, 2, 1)
Conv-256, IN, Leaky ReLU	(4, 1, 1)
Conv-1	(4, 1, 1)

IN denotes instance normalisation.

Table B.3
Dilated CNN-based PatchGAN discriminator.

Layer	Conv parameters (kernel size, stride, dilation)
Conv-32, Leaky ReLU	(4, 2, 1)
Conv-64, IN, Leaky ReLU	(4, 2, 2)
Conv-128, IN, Leaky ReLU	(4, 2, 2)
Conv-256, IN, Leaky ReLU	(4, 1, 2)
Conv-1	(4, 1, 1)

IN denotes instance normalisation.

Appendix C. Loss functions used in fluvial GAN and FluvialGAN_3DR

Fluvial GAN and FluvialGAN_3DR use the same loss functions except for the centre value of the gradient penalty term for the discriminator.

The loss functions are given by,

$$Loss = \begin{cases} Loss_D + GP \\ Loss_G \end{cases} \quad (5)$$

where $Loss_D$ and $Loss_G$ are hinge losses for the discriminator and the generator (Lim and Ye, 2017), respectively, and GP is the gradient penalty.

The equations of $Loss_D$ and $Loss_G$ are given by,

$$Loss_D = \mathbb{E}_{x \sim P_x} [\max(0, 1 - D(x))] + \mathbb{E}_{z \sim P_z} [\max(0, 1 + D(G(z)))] \quad (6)$$

$$Loss_G = -\mathbb{E}_{z \sim P_z} [D(G(z))] \quad (7)$$

where $G(\cdot)$ and $D(\cdot)$ are the sequential calculations of the generator and the discriminator, and $\mathbb{E}[\cdot]$ is the mathematical expectation.

The gradient penalty GP is given by,

$$GP = \lambda \mathbb{E}_{\tilde{x}} [(\|(\nabla D)_{\tilde{x}}\| - c)^2] \quad (8)$$

where λ is a constant weight of gradient penalty, $\mathbb{E}[\cdot]$ is the mathematical expectation, ∇D is the gradient of the discriminator parameters, \tilde{x} is a linear interpolation between real and fake data, and c is the centre value that equals to zero and one in the zero-centred gradient penalty (Thanh-Tung et al., 2019) and the one-centred gradient penalty (Gulrajani et al., 2017) respectively.

References

- Arjovsky, M., Chintala, S., Bottou, L., 2017. Wasserstein generative adversarial networks. In: International Conference on Machine Learning. PMLR, pp. 214–223.
- Box, G.E., Jenkins, G.M., Reinsel, G.C., Ljung, G.M., 2015. Time Series Analysis: Forecasting and Control. John Wiley & Sons.
- Brock, A., Donahue, J., Simonyan, K., 2018. Large scale GAN training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096.
- Chan, S., Elsheikh, A.H., 2019. Parametric generation of conditional geological realizations using generative neural networks. Comput. Geosci. 23 (5), 925–952.
- Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T., 2015. Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2625–2634.
- Gagniuc, P.A., 2017. Markov Chains: From Theory to Implementation and Experimentation. John Wiley & Sons.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. Adv. Neural Inf. Process. Syst. 27.
- Grimaud, J.-L., Ors, F., Lemay, M., Cojan, I., Rivoirard, J., 2022. Preservation and completeness of fluvial meandering deposits influenced by channel motions and overbank sedimentation. J. Geophys. Res.: Earth Surf. e2021JF006435.
- Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C., 2017. Improved training of wasserstein gans. Adv. Neural Inf. Process. Syst. 30.
- Heller, P.L., Paola, C., 1996. Downstream changes in alluvial architecture; an exploration of controls on channel-stacking patterns. J. Sediment. Res. 66 (2), 297–306.
- Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.
- Hoogeboom, E., Gritsenko, A.A., Bastings, J., Poole, B., Berg, R.v.d., Salimans, T., 2021. Autoregressive diffusion models. arXiv preprint arXiv:2110.02037.
- Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A., 2017. Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1125–1134.
- King, P., 1990. The connectivity and conductivity of overlapping sand bodies. In: North Sea Oil and Gas Reservoirs—II. Springer Netherlands, Dordrecht, pp. 353–362.
- Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- Laloy, E., Héroult, R., Jacques, D., Linde, N., 2018. Training-image based geostatistical inversion using a spatial generative adversarial neural network. Water Resour. Res. 54 (1), 381–406.
- Larue, D.K., Hovadik, J., 2006. Connectivity of channelized reservoirs: a modelling approach. Pet. Geosci. 12 (4), 291–308.
- Lim, J.H., Ye, J.C., 2017. Geometric gan. arXiv preprint arXiv:1705.02894.
- Lopez, S., 2003. Modélisation de Réservoirs Chenalisés Méandriques: Une Approche Génétique et Stochastique (Ph.D. thesis). École Nationale Supérieure des Mines de Paris.
- Menick, J., Kalchbrenner, N., 2018. Generating high fidelity images with subscale pixel networks and multidimensional upscaling. arXiv preprint arXiv:1812.01608.
- Mirza, M., Osindero, S., 2014. Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784.
- Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al., 2016. Conditional image generation with pixelcnn decoders. Adv. Neural Inf. Process. Syst. 29.
- Park, T., Liu, M.-Y., Wang, T.-C., Zhu, J.-Y., 2019. Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2337–2346.
- Qian, N., 1999. On the momentum term in gradient descent learning algorithms. Neural Netw. 12 (1), 145–151.
- Radford, A., Metz, L., Chintala, S., 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.
- Rumelhart, D.E., Hinton, G.E., Williams, R.J., et al., 1985. Learning internal representations by error propagation.
- Russell, C.E., 2017. Prediction of Sedimentary Architecture and Lithological Heterogeneity in Fluvial Point-Bar Deposits (Ph.D. thesis). University of Leeds.
- Slingerland, R., Smith, N.D., 2004. River avulsions and their deposits. Annu. Rev. Earth Planet. Sci. 32, 257–285.
- Snieder, S., Griffiths, C.M., Owen, A., Hartley, A.J., Howell, J.A., 2021. Stratigraphic forward modelling of distributive fluvial systems based on the Huesca System, Ebro Basin, northern Spain. Basin Res. 33 (6), 3137–3158.
- Song, S., Mukerji, T., Hou, J., 2021a. Bridging the gap between geophysics and geology with generative adversarial networks. IEEE Trans. Geosci. Remote Sens. 60, 1–11.
- Song, S., Mukerji, T., Hou, J., 2021b. GANSim: Conditional facies simulation using an improved progressive growing of generative adversarial networks (GANs). Math. Geosci. 1–32.
- Song, S., Mukerji, T., Hou, J., Zhang, D., Lyu, X., 2022. GANSim-3D for conditional geomodeling: Theory and field application. Water Resour. Res. 58 (7), e2021WR031865.
- Sun, C., Demyanov, V., Arnold, D., 2022. GAN River-I: A process-based low NTG meandering reservoir model dataset for machine learning studies. Data Brief 108785.
- Sun, C., Demyanov, V., Arnold, D., 2023. Geological realism in Fluvial facies modelling with GAN under variable depositional conditions. Comput. Geosci. 1–19.
- Thanh-Tung, H., Tran, T., Venkatesh, S., 2019. Improving generalization and stability of generative adversarial networks. arXiv preprint arXiv:1902.03984.
- Villamizar, C.A., Hampson, G.J., Flood, Y.S., Fitch, P.J., 2015. Object-based modelling of avulsion-generated sandbody distributions and connectivity in a fluvial reservoir analogue of low to moderate net-to-gross ratio.
- van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Guillard, E., Yu, T., the scikit-image contributors, 2014. scikit-image: image processing in Python. PeerJ 2, e453. <http://dx.doi.org/10.7717/peerj.453>.
- Wang, T.-C., Liu, M.-Y., Zhu, J.-Y., Tao, A., Kautz, J., Catanzaro, B., 2018. High-resolution image synthesis and semantic manipulation with conditional gans. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8798–8807.
- Williams, R.J., Zipser, D., 1989. A learning algorithm for continually running fully recurrent neural networks. Neural Comput. 1 (2), 270–280.
- Yang, Z., Chen, Q., Cui, Z., Liu, G., Dong, S., Tian, Y., 2022. Automatic reconstruction method of 3D geological models based on deep convolutional generative adversarial networks. Comput. Geosci. 1–16.
- Zhang, T.-F., Tilke, P., Dupont, E., Zhu, L.-C., Liang, L., Bailey, W., 2019. Generating geologically realistic 3D reservoir facies models using deep learning of sedimentary architecture with generative adversarial networks. Pet. Sci. 16 (3), 541–549.