

A balanced watershed decomposition method for rain-on-grid simulations in HEC-RAS

Sleimane Hariri, Sylvain Weill, Jens Gustedt, Isabelle Charpentier

▶ **To cite this version:**

Sleimane Hariri, Sylvain Weill, Jens Gustedt, Isabelle Charpentier. A balanced watershed decomposition method for rain-on-grid simulations in HEC-RAS. Journal of Hydroinformatics, 2022, 24 (2), pp.315-332. 10.2166/hydro.2022.078 . hal-03561729v2

HAL Id: hal-03561729
https://hal.science/hal-03561729v2

Submitted on 8 Feb 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Uncorrected Proof

Journal of Hydroinformatics

Check for updates

© 2022 The Authors

Journal of Hydroinformatics Vol 00 No 0, 1 doi: 10.2166/hydro.2022.078

A balanced watershed decomposition method for rain-on-grid simulations in HEC-RAS

Sleimane Hariri [a,*], Sylvain Weill[b], Jens Gustedt [a,c] and Isabelle Charpentier [a]

[a] ICube UMR 7357, Université de Strasbourg and CNRS, 67412 Illkirch, France
[b] ITES, UMR 7063, Université de Strasbourg, CNRS and ENGEES, 67084 Strasbourg, France
[c] INRIA, France
*Corresponding author. E-mail: shariri@unistra.fr

SH, 0000-0001-9079-8236; JG, 0000-0003-0779-1944; IC, 0000-0001-9079-8236

ABSTRACT

Rain-on-grid simulations for the modeling of 2D unsteady flows in response to precipitation input are implemented in Hydrologic Engineering Center's River Analysis System (HEC-RAS) software by means of a finite volume method and a sparse parallel linear solver running on a unique processor. Obviously, this may fail due to memory shortage when the discretization yields too large linear systems. Such simulations are good candidates for the design of partition and domain decomposition methods for the modeling of very large catchments by means of hydrological sub-units. Thinking in load-balanced computations, the proposed area-balanced partition method is based on the flow accumulation information. Implemented in HEC-RAS, the domain decomposition method comprises 2D–1D models combining 2D sub-unit meshes to 1D channels for extracting outflow data from upstream units and passing the information to downstream units. The workflow is automated using the HEC-RAS Controller. As a case study, we consider the French Saar catchment (1,747 km^2). The new partition methods demonstrate a better equilibrium for sub-basin areas and the computational load for the sub-problems. The total computational time is substantially reduced. Domain decomposition is required for carrying rain-on-grid simulations over the Saar watershed (2,800,000 elements). The resulting discharge and flood maps show very good agreements with available data.

Key words: area balancing, hydrological models, parallel computing, partial differential equations, rain-on-grid, watershed partitioning

HIGHLIGHTS

- The area-balanced partition method carried out the flow accumulation information.
- The load-balanced domain decomposition method using HEC-RAS for the sub-domains.
- Automation based on 2D–1D models and HEC-RAS Controller.
- Validation on 2D unsteady rain-on-grid simulations.

INTRODUCTION

Since the mid-eighties (Yang *et al.* 1998; Rodriguez *et al.* 2008), nonlinear partial differential equations (PDEs) and additional physical parameterizations are being considered to model more and more complex hydrological processes by using an increasingly accurate representation of topography. One-dimensional river models were replaced by two-dimensional (2D) hydrodynamic models operating in the riverbed. Recently, rain-on-grid software accounts for precipitation over the watershed as a source term. Best practices of hydrological models are concerned with the management of water resources, the forecasting of floods, the assessment of hazards (Falter *et al.* 2016; Fan *et al.* 2016), the simulation of extreme event scenarios (Sorribas *et al.* 2016), and landscape modifications (Bayer & Collischonn 2013), for instance.

Rain-on-grid software implements the solution of the nonlinear PDEs by means of a finite element or a finite volume method, for instance, and some structured or unstructured mesh that describes the watershed topography. Computing time and memory usage may be important as they depend on the catchment surface, the mesh size, and the stiffness of the nonlinear discrete equations to be solved in an iterative manner. A review (David & Schmalz 2021) of software, models, and parameterizations based on 20 very different rain-on-grid studies reports simulations comprising mesh elements ranging from 5,000 to 3,000,000 and run on appropriate (personal or parallel) computers. For the sake of generality, simulations are carried out by means of the 'Hydrologic Engineering Center's River Analysis System' (HEC-RAS) in the paper.

This is an Open Access article distributed under the terms of the Creative Commons Attribution Licence (CC BY 4.0), which permits copying, adaptation and redistribution, provided the original work is properly cited (http://creativecommons.org/licenses/by/4.0/).

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

Two computing strategies may be implemented to balance the computational load over a multi-core computer. On the one hand, a sparse parallel solver may be implemented as in the HEC-RAS software (Brunner 2016) to split the linear system into small blocks in order to distribute them on different cores. This approach is particularly well suited to a shared memory multi-core processor. On the other hand, a domain decomposition method (DDM) may be set up on a partition of the catchment. GIS tools are used to split the watershed into a number of hydrological sub-units (Apostolopoulos & Georgakakos 1997; Vivoni *et al.* 2011). The PDE problem is decomposed into a number of boundary value sub-problems accordingly. A DDM then organizes the solution by gathering the solutions computed for the sub-units and by setting relevant boundary conditions at the interfaces. This approach is efficient on a shared memory multi-core processor and on a multi-processor cluster, assuming a good balance of the computational effort.

These computing strategies can be combined. A watershed partition is needed to implement a DDM. Then, boundary values corresponding to each of the sub-domains can be solved using a sparse parallel solver. The efficiency of this computing approach depends on the performance of its components. In particular, the watershed partitioning method should take into account a digital elevation model (DEM) and user-prescribed points for the modeling of the topography, gauging information for validation purposes, and DDM requirements for computational performances. Given a catchment, state-of-the-art partition methods Apostolopoulos & Georgakakos (1997); Vivoni *et al.* (2011); Pontes *et al.* (2017) propose to split the stream at confluences, then with respect to the reach length, ignoring any information about the area drained by a hydrological sub-unit. In the present paper, we use this flow accumulation information to propose a balanced watershed partition and decomposition method adapted to large rain-on-grid simulations.

The paper is organized as follows. A direct rainfall model as well as terrain and methodological constraints relevant to watershed partition methods are described in the **Methodological framework** section. We evaluate two state-of-the-art techniques and their limits. The section **Area-balanced domain decomposition method** introduces the area-balanced partition method and discusses a domain decomposition method adapted to rain-on-grid modeling. Our implementation builds on the HEC-RAS software and the HEC-RAS Controller tool. Rain-on-grid simulations carried out for two large catchments are reported and discussed in the section **Results** to demonstrate the abilities and performances of the proposed methods. The last section provides **Conclusions.**

Methodological framework

The hydrological modeling of large-scale watersheds through the discretization of nonlinear PDEs usually involves a large number of variables and a long computational time.

Domain decomposition methods were proposed to face both memory and time shortages (Leandro *et al.* 2014; Kumar & Duffy 2015). Such a partition may also account for hydraulic structures that locally impact the flow in an important manner, gauging stations along the stream, or soil roughness heterogeneity, for instance.

This section introduces general equations for rain-on-grid modeling and evaluates memory requirements. State-of-the-art watershed partitioning methods are recalled to emphasize their capabilities and limitations, as well as the potential added value of such processes.

Direct rainfall model in HEC-RAS

Rain-on-grid modeling is implemented by means of nonlinear PDEs. Without limiting the general scope, we focus on the two models implemented in the version 5.0 of HEC-RAS (Brunner 2010).

The 2D shallow water equations (continuity and momentum, respectively) are written in the watershed Ω as

$$\frac{dh}{dt} + \nabla(uh) = R \tag{1}$$

$$\frac{du}{dt} + (u\nabla)u + \frac{v_t}{h}(h\nabla u) + g\nabla(h + z) = gS_f \tag{2}$$

where the unknowns h and u are the water depth and the flow velocity vector, respectively. The compound source/sink flux R allows to account for precipitation, evaporation, and infiltration terms. Other parameters are the bed elevation z, the gravitational acceleration g, the horizontal eddy viscosity coefficient v_t, and the bottom friction coefficient S_f.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

Assuming that the unsteady term, the advection term, and viscosity term can be ignored, the diffusive wave approximation of the shallow water momentum Equation (2) may be written as:

$$g\nabla(h + z) = gS_f \tag{3}$$

Continuity and momentum equations are solved together with boundary conditions set on the watershed.

$$\partial\Omega = \partial\Omega_{in} \cup \partial\Omega_{out} \cup \partial\Omega_{side} \tag{4}$$

These are used to model:

- a flow Q_{in} entering in the domain Ω at the inflow boundary $\partial\Omega_{in}$:

$$\iint u.nds = Q_{in} \tag{5}$$

where n is the unit normal vector to the boundary, s is the side boundaries, and the integral is taken over the boundary surface on which the boundary condition applies.

- an outflow condition at the outlet $\partial\Omega_{out}$ implemented as a water surface gradient S_b:

$$\nabla(h + z) \cdot n = S_b, \quad \text{on } \partial\Omega_{out} \tag{6}$$

- a normal flux condition on the side boundary Ω_{side}:

$$\nabla u \cdot n = 0, \quad \text{on } \partial\Omega_{side} = \partial\Omega \,/\, (\partial\Omega_{in} \cup \partial\Omega_{out}) \tag{7}$$

The nonlinear equations are discretized using an iterative scheme for the temporal integration and a finite element volume for the spatial component. This accounts for the watershed topography, which is provided as a DEM and a shapefile that delineates the area of interest. Other data are boundary values, Manning's roughness coefficient accounted in S_f, and hydrological data R.

It is worth mentioning that, using HEC-RAS, linear systems are solved by means of a sparse parallel solver that can afford medium-range discretization involving a few hundred thousand of elements on a shared memory multi-core processor. In other words , due to memory shortage, the HEC-RAS sparse parallel solver abilities may be insufficient to perform rain-on-grid simulation on a unique multi-core processor when the discretization involves a very large-scale watershed and/or very fine mesh. Indeed, as discussed in David & Schmalz (2021), the application of the HEC-RAS rain-on-grid for the simulation of large-scale watersheds remains limited by the number of mesh elements in the computational domain (David & Schmalz 2021). For example, the simulation of the Saar basin of 1,747 km^2 using a 25 m mesh generated about 3 million mesh elements, which exceeds the capacity of the available memory of the Intel Xeon 64 bits @3.80 GHz 6-core processor with 16 Gb of RAM memory we used.

In a nutshell, HEC-RAS 2D is mainly used on Windows platforms and the set-up of a multidomain simulation remains challenging. In this paper, a proof of concept is proposed to evaluate the benefits of the domain decomposition method. It is worth mentioning that HEC-RAS neither allows to delineate watersheds nor performs a partition at confluences in an automatic manner.

Stream-based methods

Provided with a DEM and an outlet, GIS software tools such as ArcGis, QGIS, or TopoToolbox (Schwanghart & Kuhn 2010) allow for the delineation of the corresponding watershed and stream network through the calculation of a flow direction and a flow accumulation. As an illustration, the flow accumulation raster related to the case study DEM is plotted in Figure 1. It records the number of cells, and, thus, the area drained by each of the DEM cells, outlining the stream network. The latter comprises a number of connection points called confluences and a number of links called river reaches.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

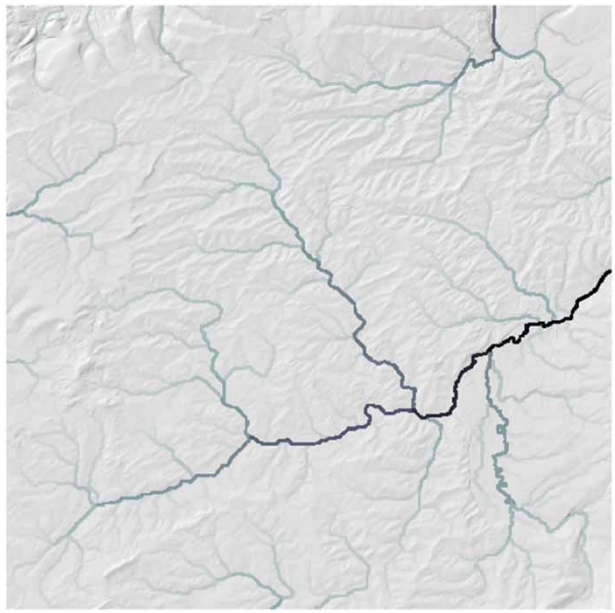


Figure 1 | Flow accumulation raster for the case study DEM, see Results.

Based on the confluence points, the basic watershed partition offered by GIS tools allows to delineate sub-basins. When the stream network presents nearby confluences, this partition method may create very small sub-basins. On the other hand, long reaches may result in large sub-basins.

As proposed in Pontes *et al.* (2017) and implemented in IPH-Hydro Tools (Siqueira *et al.* 2016), the confluence method may be improved by dividing the longest river reaches to delineate smaller sub-basins of larger ones. However, this does not remedy the problem of very small sub-basins.

The former two methods may be viewed as zero-dimensional and one-dimensional since they are based on confluence points and reach lengths, respectively. This stream information cannot guarantee an area-balanced partition of the watershed, because the available flow accumulation information is not taken into account. This will be illustrated for a case study in the section **Numerical results**.

Added value of a partition method

DEMs are only an approximation of the topography and may underestimate the significant impact of bridges and other engineered structures on the water flow.

Watershed partitions can account for civil engineering structures by setting inlets/outlets that are close to them or by comparing the computed and observed discharges when a gauging device is available. A partition also allows to represent the heterogeneity of the terrain and the meteorological variability over the catchment.

From a computational point of view, finite element models that are applied to sub-units with a similar area involve linear systems with a similar number of unknowns. Thereby, their solution has a comparable computational cost, and the domain decomposition method is more efficient (Kumar & Duffy 2015).

The partition method we propose in the section **Area-balanced domain decomposition method** aims to:

1. account for engineered structures as user-defined outlets,
2. consider small, but homogeneous hydrological sub-units,
3. ensure a balance of the sub-unit areas,
4. obtain a prescribed number of sub-units, and
5. perform simulations with a reasonable number of unknowns per sub-unit.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

The first two properties ensure a better quality of the modeling. Properties 3 and 4 improve the efficiency of the simulation. The last enables the opportunity for rain-on-grid simulations on very large watersheds. The last three properties are not satisfactorily addressed by previous methods as reported in the section **Results**.

AREA-BALANCED DOMAIN DECOMPOSITION METHOD

DEM analysis and sub-unit delineation

GIS are precious tools to support hydrological modeling. Software like ArcGIS or QGIS provides graphical user interfaces and programming interfaces with Python. In contrast to this, TopoToolbox (Schwanghart & Kuhn 2010) is a set of Matlab functions dedicated to DEM analysis, with a special focus on hydrology. An overview of the respective performance of these tools is proposed in Schwanghart & Scherler (2014). Among them, we choose TopoToolbox because it is integrated in a scientific computation framework with good efficiency.

TopoToolbox is an object-oriented toolbox provided with a user guide that describes the methods that are necessary to our partition tool. As figured out in listing 1, the DEM is read as a grid object. The flow direction and flow accumulation rasters are then computed to deduce the stream network as an ordered list of cells that drained a minimum number of cells and drainage basins using or not input information (prescribed points, for instance). The minimum drained cells threshold is denoted by Delta.

Listing 1: Drainage basin delineation.

```
dem    = GRIDobj ('DEM - file.asc');
floDi = FLOWobj (dem, 'preprocess','carve');
floAc = flowacc (floDir);
stream = STREAMobj (floDir, floAcc > Delta);
db     = drainagebasins (floDir, input);
```

State-of-the-art partition methods are based on the split method operated, first, with respect to stream confluences. The partition with respect to a prescribed reach length has been implemented using the distance property of the stream object to identify complementary points. These partition methods are exemplified and analyzed in the section **Results**.

Partition under constraints

Targeting rain-on-grid simulations for large-scale watersheds has motivated the design of the new partitioning method. To accommodate parallel computing in the near future, a special emphasis is put on area balancing by considering the flow accumulation raster (2D information) rather than the reach length (1D information) or the confluence information (pointwise information).

Put briefly, for each cell of the flow accumulation raster, we maintain the number of upstream cells it drains. This parameter is, thus, related to the area of its upstream watershed.

The workflow is illustrated in Figure 2. A preprocessing is set up to read the DEM and, possibly, user-defined points that represent the outlets and engineered structures of interest.

The area-balancing watershed decomposition method is organized as follows. The sub-unit with the largest area is selected (imax) and its flow accumulation information is computed. A splitting point is determined by searching for the cell that drains half of that flow (coor(iSplitPt); see Figure 3). This is added to the set of outlet points defining the drainage basins. This process is iterated until the desired stopping criterion is met. This can be based on the number of sub-units and/or on the sub-unit areas. In listing 2, the process stops when all the sub-basins have an area less than a prescribed maxArea.

Listing 2: Pseudo-code for the area-balanced partition method.

```
outletPts = <user - def outlets >;
db = drainagebasins (floDir, outletPts);
[maxSubA, imax] = polyarea (db);
while  (maxSubA > maxArea)
        subFloAcc = flowacc(db(imax));
        [halFA, iSplitPt] = max(subFloAcc)/2;
        [outletPts] = [outletPts coor(iSpliPt)];
        Db = drainagebasins (floDir, outletPts);
        [maxSubA, imax] = polyarea (db);
```

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

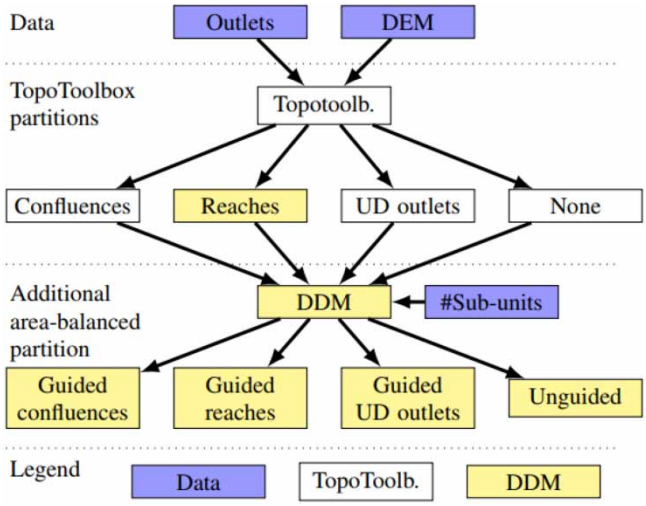


Figure 2 | Partitioning workflow.

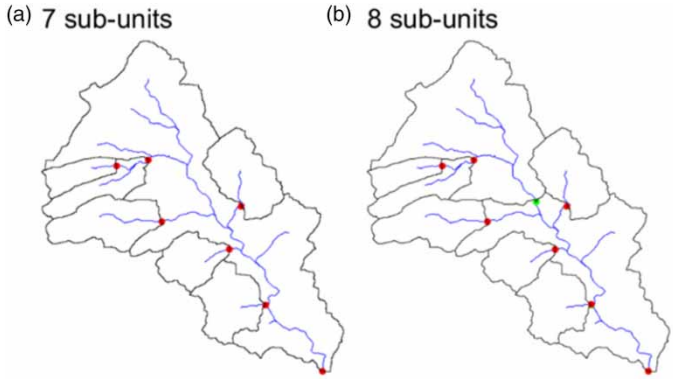


Figure 3 | Guided partitioning. (a) User-defined points (red dots) and (b) partition of the largest watershed into two sub-watersheds of a similar area (green dot).

The sub-watersheds are saved in the shapefile format using the shapewrite function of the Matlab Mapping toolbox such that it then can be used in the HEC-RAS simulations, for instance.

Confluence method unguided method

Depending on the actual topography or its digitization, the watershed delineation may produce sub-units that present isolated cells, notably near to the outlet; see Figure 4. Defects with respect to the 4-neighbor connectivity may occur for any partition method or partition software. These are not appropriate to a finite element or a finite volume modeling.

At the end of the partition process, a connectivity check may be performed to correct the decomposition by moving some of the outlet points upward. This can be achieved automatically by inspecting the properties of each sub-unit by means of the regionprops Matlab function, for instance. It is worth noticing that the user can also take them into account in a channel, as proposed in the 2D–1D modeling described hereafter.

Domain decomposition with HEC-RAS

The RAS Mapper interface is used to model the terrain starting from a DEM as follows.

1. Sub-unit shapefiles are imported to implement an area-balanced domain decomposition method.
2. Each sub-unit is meshed to discretize the 2D PDEs, either (1)–(2) or (1) and (3).

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

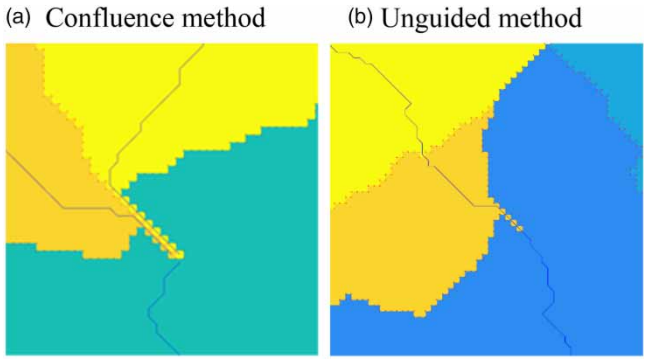


Figure 4 | Isolated cells near to outlets. (left) Partition method at confluences and (right) unguided area-balanced partition method.

3. A 1D river model is added at the sub-unit outlet to extract the output flow hydrograph (Equation (6)) as proposed in Goodell (2018).
4. This flow hydrograph serves as an input to the downstream sub-unit following Equation (5).

Note that this 2D–1D modeling of the sub-unit allows for the automation by means of the HEC-RAS Controller (Goodell 2014) that has been designed for 1D problems.

In a nutshell, the HEC-RAS Controller provides a collection of routines that allows to open a project, to select and to run the so-called plans, and to extract output data.

The DDM sequence and the communications are then automated by means of a Virtual Basic for Applications (VBA) driver written using the HEC-RAS Controller functions.

The VBA script, see Figure 5, organizes the computation sequence by accounting for the natural water flow. It runs sub-unit models from the most upstream to accumulate the stream flow at the watershed outlet. Each sub-unit calculation is carried out in one go, providing all the necessary input flow data to its downstream sub-unit. The number of HEC-RAS plans corresponds to the number of sub-units.

RESULTS

The performances of the partition and domain decomposition methods are compared on two case studies: the Saar watershed in France (1,747 km^2) and the Moderbach sub-watershed (89 km^2). These are evaluated and analyzed on the generation of nested partitions (Case study 1) and the solution of 2D unsteady rain-on-grid simulations (Case study 2). HEC-RAS version 5.0.7 is used to model the hydrological behavior where the catchments are discretized using a mesh size of 25 m.

Run-times were recorded in minutes on a Intel Xeon 64 bits @3.80 GHz 6-core processor with 16 Gb of RAM memory. The sparse parallel solver of HEC-RAS makes use of the six cores of that processor.

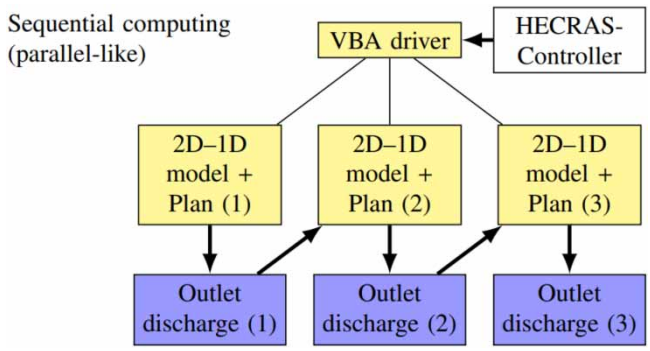


Figure 5 | DDM/HEC-RAS workflow.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

Validation and performances on a small watershed

Case study 1

The slightly undulating catchment of the Moderbach stream (89 km^2, Région Grand-Est, France) is represented in Figure 6. Elevations range from 210 to 305 meters above sea level (m.a.s.l). The DEM raster is issued from BD Alti®25 m – V2 and is available for research purposes.

Due to the clayey soil, the watershed was converted into a part of the Maginot defense waterline in the 1930s, comprising 11 dams delineating six reservoirs on the Moderbach's tributaries (blue polygons) and five forebays along the main stream delineated by five dams (red lines) to be flooded. These structures that may strongly impact the flow are used to compare guided and unguided area-balanced partition methods.

This HEC-RAS version ignores infiltration. Such a strong hypothesis is plausible for this watershed since soils are quasi-impervious under winter conditions.

Comparison of partition methods

Figure 7 plots partitions computed with the two state-of-the-art methods (confluences, then a reach length of 2 km between two outlets) and the unguided area-balanced partitions.

The characteristics of the partitions are reported in Table 1. On the one hand, the confluence method and the reach length method produce very small sub-watersheds. In both cases, the ratio between the maximum and the minimum area is larger than 20. The balance of areas is not satisfactory. On the other hand, the ratio of 2 obtained for the novel partition method (U. Area) demonstrates the balance of the sub-watershed areas when similar mesh sizes are used for the different sub-units.

The computation time for the confluence method corresponds to the time needed to proceed the DEM following the listing 1. The time measured for the unguided area-balanced method comprises those of listing 1 and 2. Note that this difference of a few seconds is very small as compared to the computation time needed for rain-on-grid simulations.

Confluence method. The stream network is a set of connected segments, the number of which is parameterized by means of a threshold Δ on minimum drained cells, see Listing 1. The characteristics of the nested partitions produced with the thresholds are reported in Table 2.

As the number of confluences increases, there is an increasing probability for a very small sub-unit in the partition. Consequently, the ratio between the maximum and the minimum areas increases.

Guided area-balanced method. The guided method is carried out by starting from the seven user-defined outlets that are located at the pond outlets and the watershed outlet. The generated sequence of nested partitions is illustrated in Figure 8,

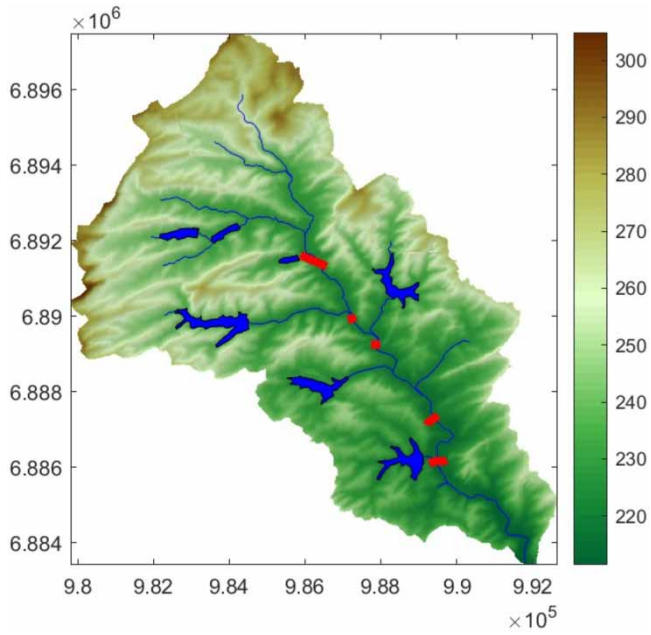


Figure 6 | DEM of the Moderbach stream (EPSG:2154).

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

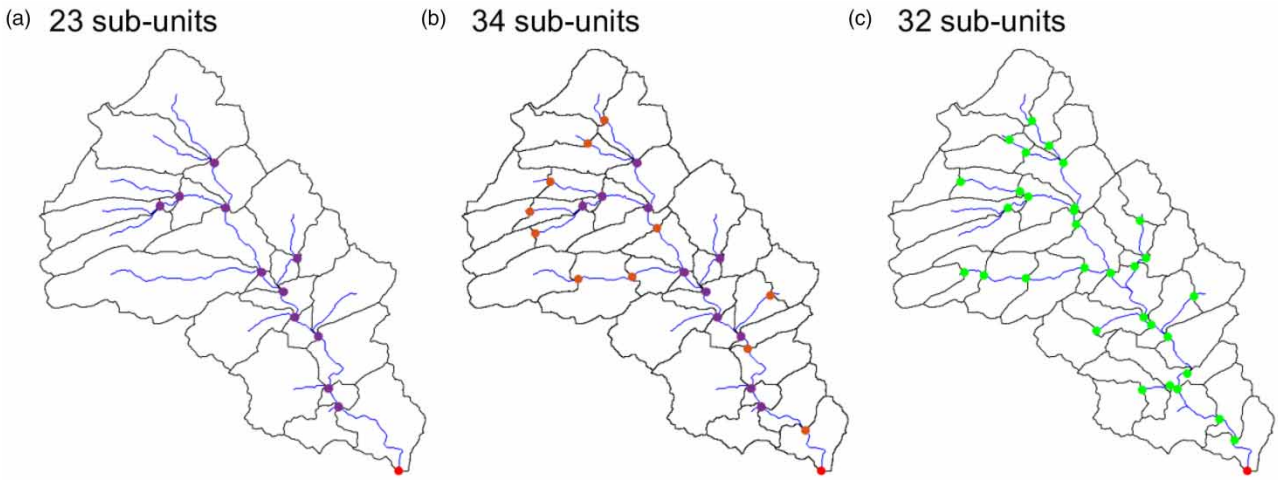


Figure 7 | (a) Partition at confluences (magenta dots), (b) additional partition with respect to a reach length of 2 km (orange dots) and (c) unguided area-balanced partition (green dots).

Table 1 | Characteristics of the partitions displayed in Figure 7 and computation times

Properties	(a) Confluence	(b) Reach	(c) U. Area
#Sub-units	23	34	32
Max. area	11.5	6.7	4.0
Min. area	0.2	0.2	2.0
Ratio	47.5	27.7	2.0
Mean area	3.8	2.6	2.8
SD	2.7	1.5	0.6
Time (s)	4.2	10.9	16.9

Areas are expressed in km^2.

Table 2 | Characteristics of the confluence partitioning method and computation times

Properties	Confluence partitions			
Δ	20,000	15,000	9,200	3,100
#Sub-units	3	5	9	23
Max. area	58.1	42.0	17.4	11.5
Min. area	13.6	4.9	0.4	0.2
Ratio	4.3	8.6	37.0	47.5
Mean area	29.7	17.8	9.9	3.8
SD	24.9	14.2	5.6	2.7
Time (s)	3.7	3.8	3.9	4.2

Area is expressed in km^2.

and partition properties are reported in Table 3. One observes that the ratio decreases from 19.3 corresponding to the user-defined partition to 9.1 because the partitions into 7 and 8 sub-units involve very large sub-units.

Unguided area-balanced method. The unguided method only requires the watershed outlet as an input. The nested partitions into a number of sub-units (that can be a power of 2) are plotted in Figure 8. Corresponding properties are

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

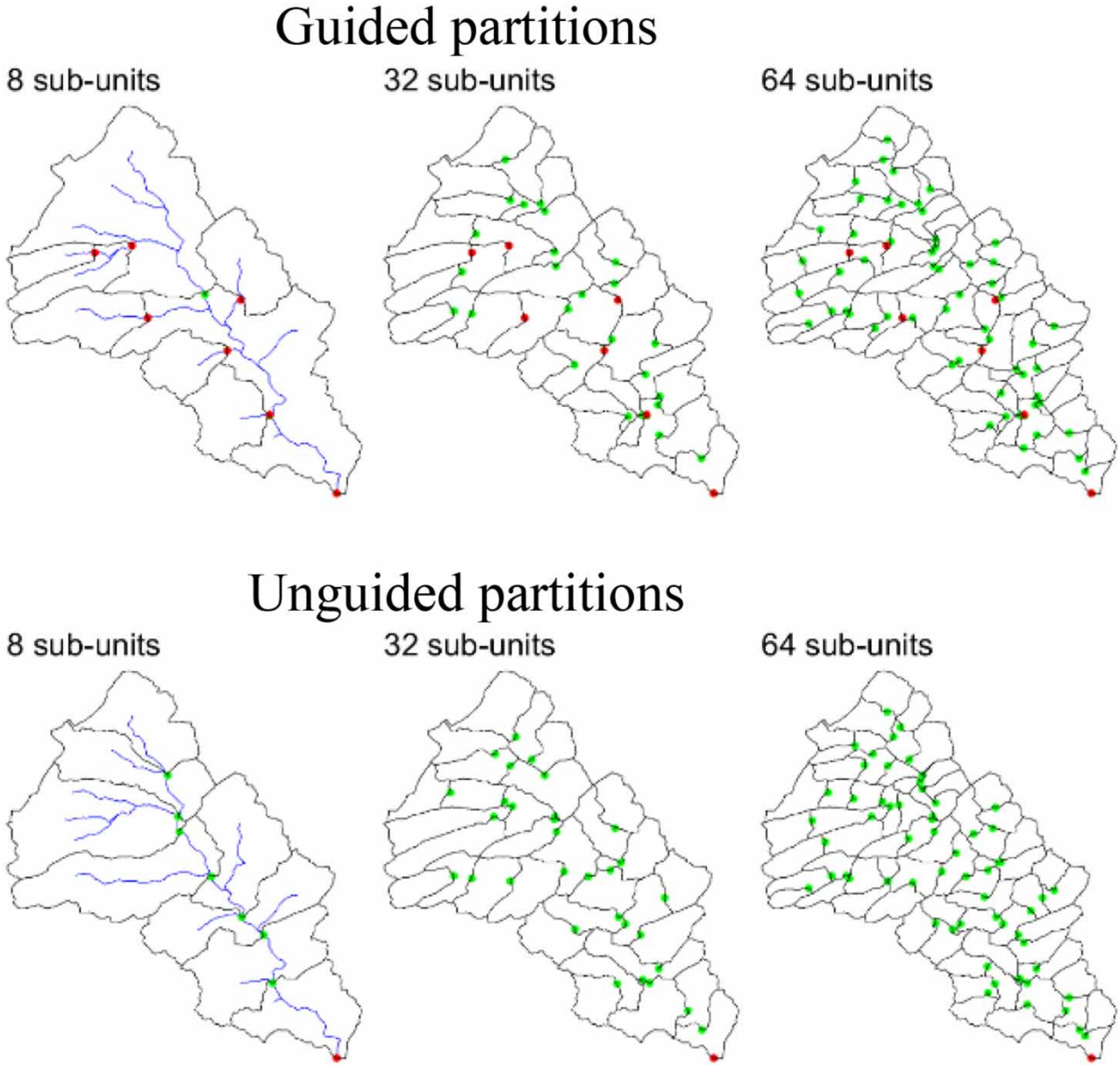


Figure 8 | Some guided and unguided iterative area-balanced partitions.

Table 3 | Characteristics of the guided area-balanced partitioning method and computation times

Properties	Partitions from user-defined points				
#Sub-units	7	8	16	32	64
Max. area	57.9	27.4	7.6	3.9	2.0
Min. area	3.0	3.0	2.9	1.7	0.7
Ratio	19.3	9.1	2.6	2.3	2.9
Mean area	12	11.1	5.6	2.7	1.4
SD	18.7	10.2	1.6	0.6	0.3
Time (s)	4	6.2	9.5	14.9	26.7

Area is expressed in km^2.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

Table 4 | Characteristics of the unguided area-balanced partitioning method and computation time

Properties	Unguided partitions					
#Sub-units	2	4	8	16	32	64
Max. area	47.1	30.0	15.8	8.1	4.0	2.0
Min. area	42.1	17.1	8.1	3.8	2.0	0.7
Ratio	1.1	1.8	1.9	2.1	2.0	2.9
Mean area	44.6	22.3	11.1	5.6	2.8	1.4
SD	3.5	5.5	2.7	1.3	0.6	0.3
Time (s)	6.2	7.2	8.9	11.4	16.9	28.9

Area is expressed in km^2.

reported in Table 4. This shows an increase of the ratio from 1.1 to approximately 2.9 since the first two sub-units have almost the same area.

To summarize, the two area-balanced methods provide similar performances when the number of sub-units is large. It should be mentioned that in these two methods, the minimum drained cell threshold Δ is set to 1,000 allowing to reach a large number of sub-units. The unguided method outperforms the confluence method in any case.

Domain decomposition results

The catchment is partitioned, then meshed using about 14,5703 elements. The domain decomposition performances are evaluated by simulating a precipitation event of 28 h with an amount of 80 mm/day to saturate the watershed in a reasonable time. Additional 44 h are carried out with no precipitation in order to drain the watershed until reaching a null discharge at the outlet. The Manning's coefficient is set to 0.03. The boundary condition at the 1D river uses a normal depth estimated using the RAS Mapper tool: the slope parameter is set to 0.001. The maximum flood extent is plotted in Figure 9.

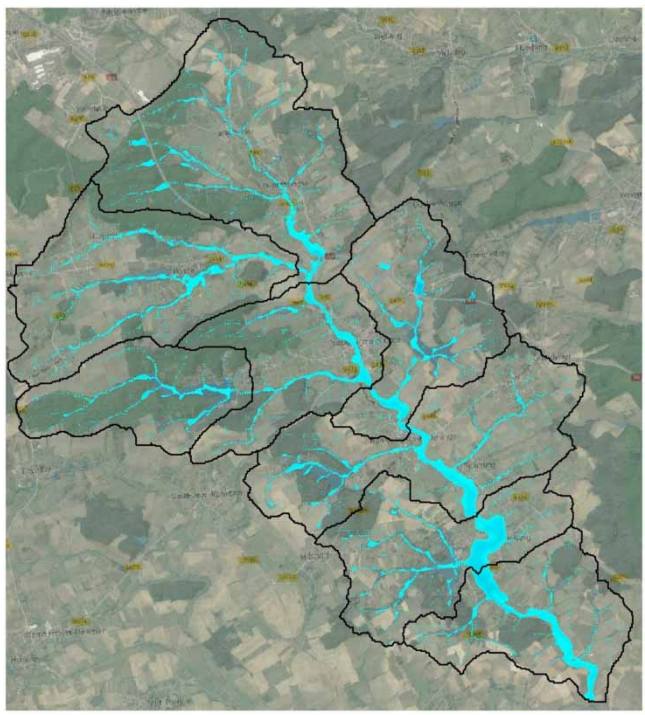


Figure 9 | Flow extent generated from the simulation of the Moderbach decomposed into eight sub-units using unguided partition.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

Presented in Figure 10(a) are the discharge results computed at the outlet with HEC-RAS and those obtained with the DDMs (workflow of Figure 5) involving nested partitions of 2, 4 and 8 sub-units, respectively. Not shown are very similar results with the confluence partitioning method.

The relative errors displayed in Figure 10(b) exhibit very small discrepancies. These may be due to the value of the slope parameters, topographical effects, and the partition method, since the null flux boundary condition (7) is applied onto the boundaries $\partial\Omega_{side}$ for the catchment and the sub-units, respectively.

The computational efficiency of the DDM is evaluated for a number of partitions carried out with respect to either the confluences or the (unguided) area balancing.

To evaluate the performances of the sparse parallel solver of HEC-RAS and the general behavior of the DDM, an overview of the computation times with respect to the sub-unit areas and the partition methods is provided in Figure 11. In Figure 11(a), points of the same color and of the same shape correspond to sub-units of the same partition. Triangles and filled circles are used to the unguided area-balanced partition method and the confluence partition method, respectively.

As expected, points are almost aligned in this log-log representation. The computation times and the areas are linked by a power law, the slope of which reflects the performance of the HEC-RAS sparse parallel solver. The differences in the solution times for the sub-units of very similar area indicate that the latter is a key parameter, but not the only one in this nonlinear solver.

Moreover, Figure 11(a) exhibits three colored clusters (filled circles of the area-balanced method) with a very similar area and computing time. In contrast to this, the dispersion of the triangular points is much higher for the confluence method and very small sub-units are created.

The total time and the speed-up are plotted, Figure 11(b) and 11(c), against the number of sub-units. We see that for both partitioning methods, the decrease in computing time is multiplicative. As we have seen, the computation of the linear system generally grows super-linear in the size of the parts; thus, with Jensen's inequality (Jensen 1906), the sum of the times of the linear systems of all parts is much smaller than for the linear system of the original watershed.

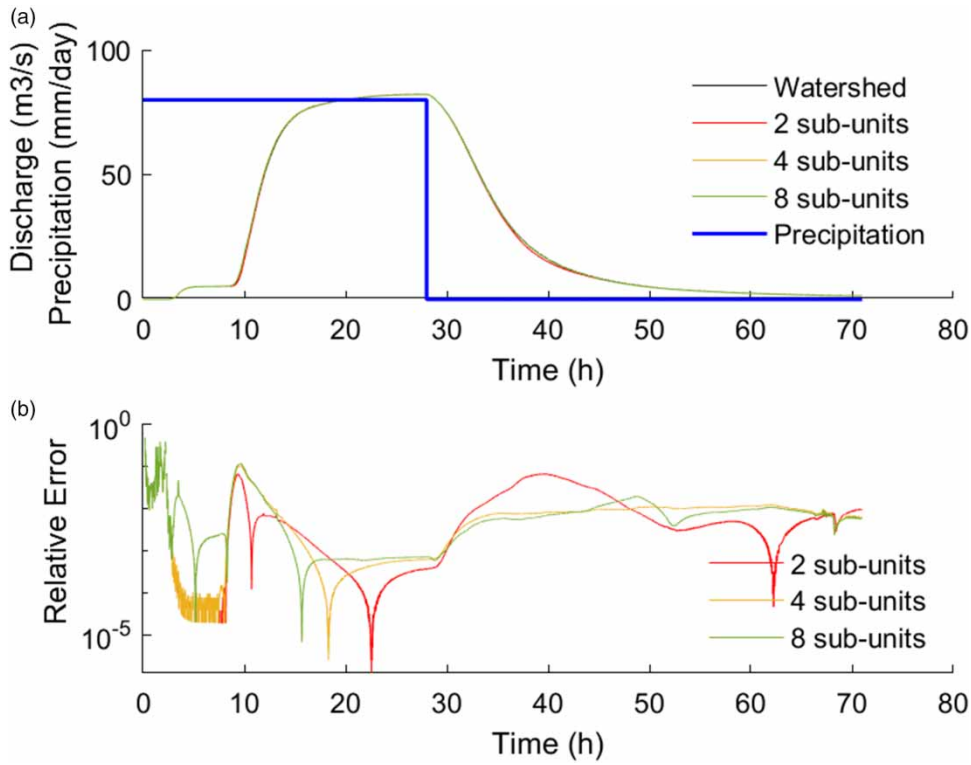


Figure 10 | Outlet flow computed using the unguided area-balanced partition. (a) Discharge and (b) relative error evaluated with respect to the discharge computed on the total watershed.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

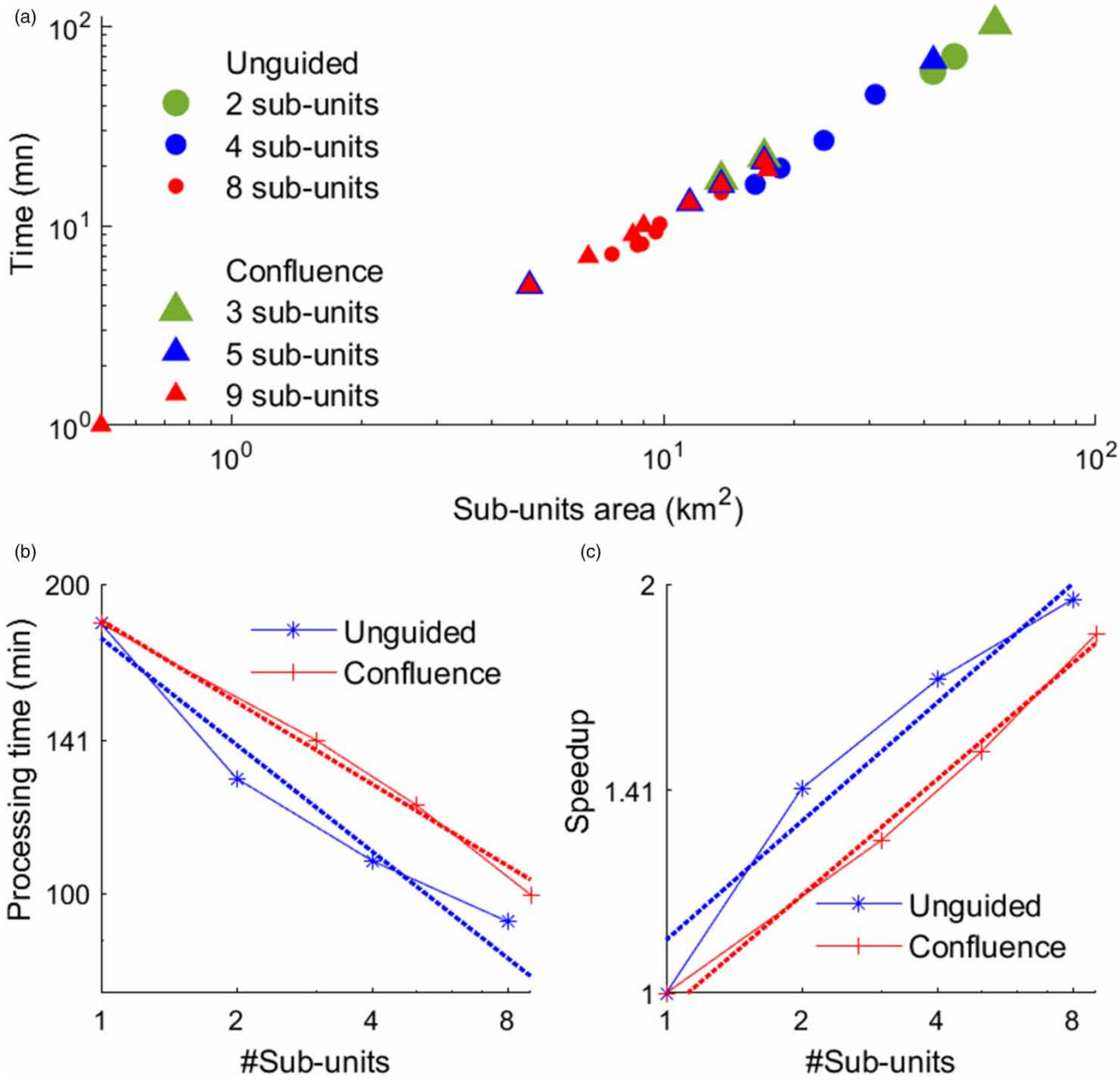


Figure 11 | DDM computing times with respect to sub-units. (a) Computation time per basin for the unguided area-balanced partition (triangles) and the confluence partition (filled circles), (b) processing time, and (c) speed-up for different partitions.

Additionally, Figure 11(b) shows regressions of the form ax^b, where b is -0.27 and -0.32, respectively. This clearly shows the advantage of the partition method by a DDM in general, but also that the unguided balanced partition method improves the confluence method by an important factor.

Rain-on-grid over a large watershed

In a first attempt, we designed a HEC-RAS plan modeling the Saar catchment in France with an element size of 25 m, i.e. involving about 2,800,000 elements, to verify the occurrence of a memory overflow. Such an issue has motivated the development of both the area-balanced partition and decomposition method.

Case study 2

Originating in the Vosges Mountains (Région Grand-Est, France), the Saar River (246 km long) drains an area of (1,747 km^2) before it enters the German territory at the confluence with the Blies River located in Sarreguemines. Elevations range from

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

800 m.a.s.l. down to 243 m.a.s.l. Twelve rasters issued from BD Alti®25 m – V2 are concatenated into a unique raster comprising 12×10^6 cells. Precipitation data are from the Kappelkinger station (Météo-France).

Eleven gauging stations are distributed along the main stream, the data of which may be found on http://www.hydro.eaufrance. Among them, we select the stations located in the two main cities, namely Sarralbe (Station A9200100) and Sarreguemines Centre (Station A9311050), to compare computed and measured discharges for the extreme flood event that occurred from 22nd December to 24th December 2010.

A flood map and the related shapefiles were produced by the SERTIT (Service régional de traitement d'images et de télédétection, Strasbourg) from SAR Cosmo-Skymed images (5 m resolution). It is worth noticing that these images cause an underestimation in urban and forested areas. Moreover, the permanent waterbodies (river and ponds) and very small water objects (less than 100 m^2) were not reproduced. The shapefile of the flood extent on 22nd December is used for comparison with our simulation results.

Simulation results

Computational performances. The catchment is split into 20 sub-basins using the guided area-balanced method (Figure 12) to account for the 11 gauging stations located on the main river (red dots). Other sub-basin outlets are represented using green dots. The sub-basins range from 11 to 164 km^2 in area (see Table 5), 17,253 to 26,2183 in mesh elements and 15 to 405 min in computational time. The simulation organized following the workflow of Figure 2 was completed in 86 h. As expected, the partition time is several orders of magnitude smaller than the simulation time.

Table 5 also reports the characteristics of an area-balanced partition method involving small sub-basins (an average area of 13.3 km^2) and a partition at the confluences, both performed from the same stream network. Here again, the ratio is in favor of the area-balanced method since the confluence method generates very small sub-units.

Hydrological results. The flood event of 22–24 December was preceded by heavy precipitation a few days before (50 mm, 6–9 December) that saturated the soil. This makes case study 2 consistent with the 2D rain-on-grid simulation in HEC-RAS that does not consider the infiltration effect.

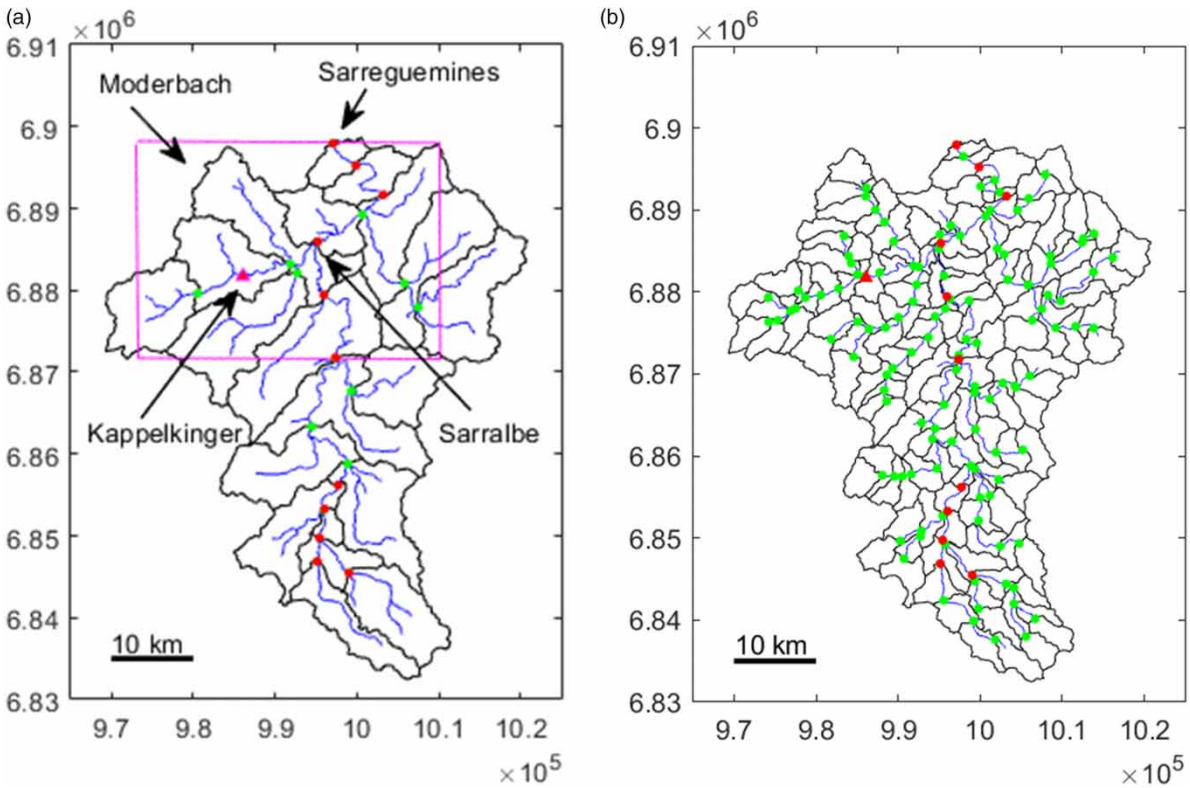


Figure 12 | Area-balanced partition of the French Saar catchment into 20 and 131 sub-basins at gauging stations (red dots) and additional points (green dots). The magenta box delineates the observation area (EPSG:2154).

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

Table 5 | Characteristics of the Saar partitions and computation times

Properties	Guided	Partitions	Confluence
#Sub-units	20	141	78
Max. area	164.2	19.1	86.4
Min. area	10.9	8.1	0.1
Ratio	14.9	2.3	678.1
Mean area	87.3	13.3	23.3
SD	38.5	2.6	19.7
Time (s)	73.1	458.9	31.0

Area is expressed in km^2.

In Figure 13, the precipitation histogram (blue bars) and the discharge data (blue and black dash-dotted lines) provided at Sarralbe and Sarreguemines gauging stations are plotted against the discharge results (blue and black solid lines) computed at the same locations. Table 6 provides complementary information on the observed and computed water volumes, Nash–Sutcliffe efficiency (NSE) criterion (Nash 1970), the root mean square errors (RMSE), and the standard deviation (SD) computed for the timeseries.

First, relative errors lower than 12% for the water volume comes in support of the quasi-imperviousness assumption during that event. Second, both NSE values are close to 1, which represents the perfect match (Nash 1970). Third, according to Singh *et al.* (2005), a ratio of RMSE/SD lower than 0.5 as in Table 6 indicates good modeling. These indicators allow to conclude to a very good agreement between the data and the calculated discharge and validate our modeling for that event.

The observed flood extent (FO) and simulated water surface (FS) are compared in Figure 14 by representing the common flood extent ($A = FO \cap FS$) in gray. The complementary sets ($B = FS \setminus FO$) and ($C = FO \setminus FS$), plotted in blue and magenta, respectively, highlight the differences. These are notably observed in coniferous/forested areas that were not analyzed with teledetection. The partition points, including the gauging stations, are plotted as dots.

DISCUSSION

This section discusses advantages, limits, and applicability of the proposed methods from the hydrological and computation point of views. Some outlooks are raised.

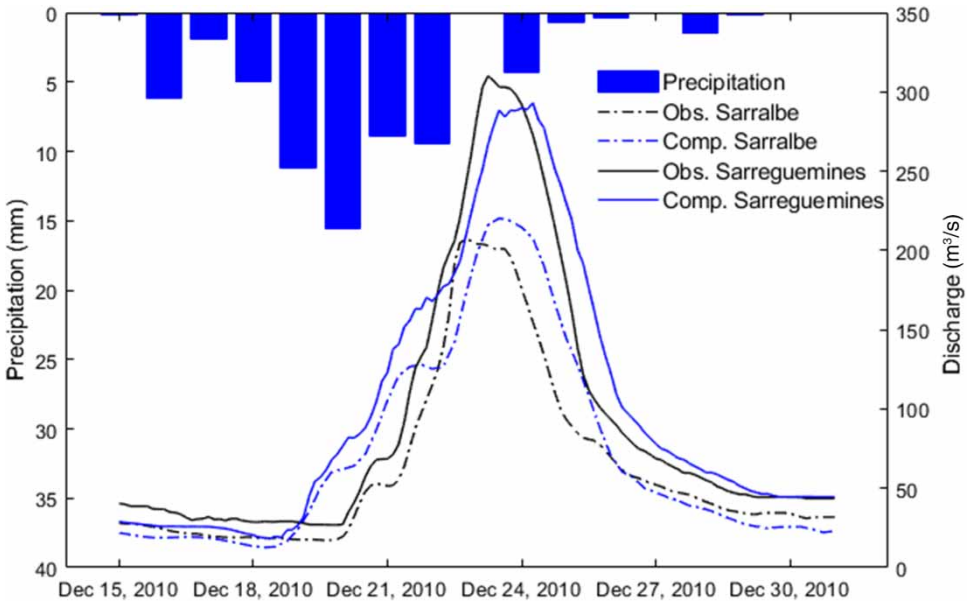


Figure 13 | Precipitation, observed, and computed discharges.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

Table 6 | Validation indicators for the rain-on-grid simulation of the December 2010 event

Sub-basin	Observed volume (m^3)	Computed volume (m^3)	Relative error (%)	NSE	RMSE	SD	Ratio
Sarralbe	24,519.9	27,856.2	11.9	0.92	25.1	56.4	0.45
Sarreguemines	35,554.2	39,095.3	9.1	0.94	26.2	84.2	0.31

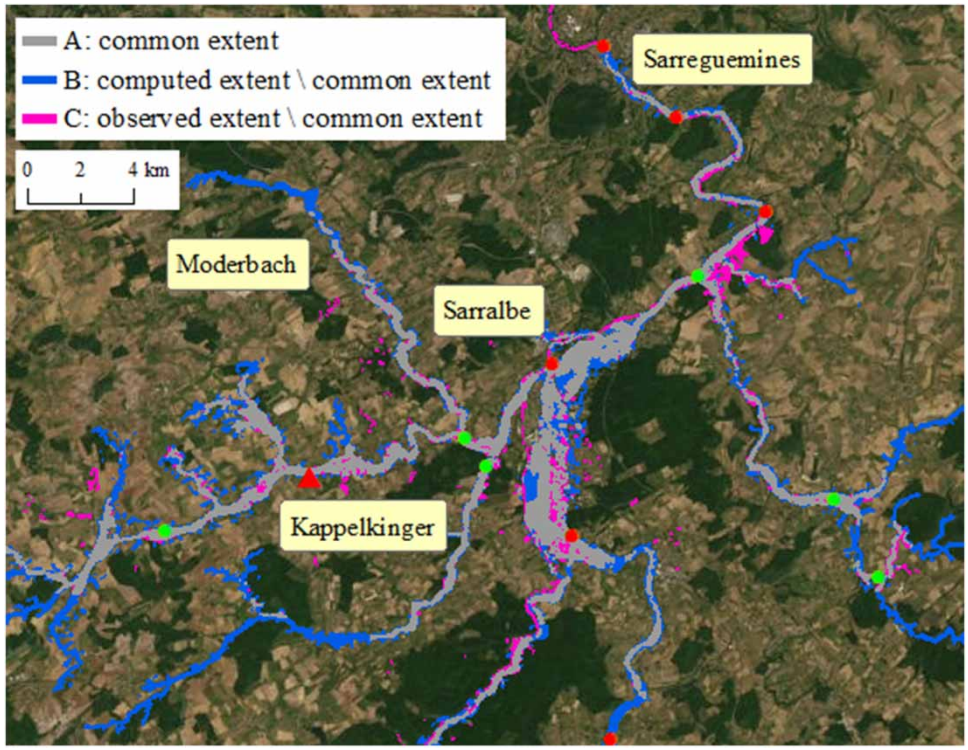


Figure 14 | HEC-RAS flood extent on the flood map of December 2010.

Hydrological vs. hydraulic modeling

Some user-defined outlets may be set to facilitate the account for civil engineering structural or topographical effects in the simulations. Complementary outlets may be positioned in an automatic manner either at confluences or at points determined with respect to particular criteria constraints without, however, fully addressing the area-balancing issue as demonstrated in Table 2.

One of the novelties of the paper is to organize this additional phase in order to delineate sub-units with a similar area using the flow accumulation information. As demonstrated in Tables 3 and 4, both versions (user-guided or unguided) of our automated area-balanced partitioning method perform much better. Moreover, these sub-units can be used with a distributed hydrological model in an efficient manner from both the computational time, see Figure 11, and the memory requirements.

The DDM method allows to build simulation plans with homogeneous data over the sub-units, but with heterogeneity over a large watershed. In particular, this may facilitate the account for more actual precipitation and soil properties (Manning's coefficient). It is worth mentioning that the Saar catchment is submitted to *Cfb* climate (Köppen classification) that is with temperate oceanic climate. In particular, winter precipitation events occur at a regional scale. Thus, heterogeneity in the precipitation amount is not implemented to carry out fair hydrological and methodological comparisons.

This HEC-RAS version ignores infiltration. As demonstrated in Figure 13, such a strong hypothesis is plausible for this watershed since soils are quasi-impervious under winter conditions. Such an assumption may also agree with surface sealing of soil in urban area (although additional factors could be considered) or dry soils. To overcome the imperviousness assumption, a SWAT/HEC-RAS integrated modeling (Zeiger & Hubbart 2021) could be considered, for instance.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

Scientific computing aspects

A 2D PDE solver is time- and memory-consuming, especially when the domain under study becomes large. This applies to the 2D unsteady rain-on-grid solver recently proposed within the HEC-RAS software. It processes linear systems by means of a sparse parallel solver to take advantage of a unique multi-core processor. This software is, thus, a perfect candidate to assess the benefits that could be brought by an additional partition of the watershed into hydrological sub-units.

The impact of the improved balance in the sub-unit areas on the 2D PDE computation is assessed using HEC-RAS in Figure 11. For our first case study, the DDM workflow described in Figure 5 resulted in a speed-up of 2 when eight sub-units of similar size are considered. Performances are a little lower with the confluence method.

For our second case study, the DDM workflow allows us to run simulation involving a larger number of unknowns (2,800,000) as compared to previous rain-in-grid simulations (David & Schmalz 2021) carried out with HEC-RAS. This is a major improvement.

Limitations and outlooks

HEC-RAS plans should be built using the graphical user interface. The multiplication of the sub-domains requires some initial manpower to set up 2D–1D models. Nevertheless, the cost–benefit ratio becomes important as soon as the rain-in-grid model is used in an operational manner and run many times as in a data assimilation process.

Local mesh refinements that could impact the load balance in a parallel implementation will be considered in the near future.

For now, the VBA code developed using the HEC-RAS Controller functions allows to automate the DDM/HEC-RAS coupling strategy. The implementation on a multi-processor cluster could bring an additional reduction of the time to results. For the moment, this strongly depends on the maximum number of the sub-units in the upstream-to-downstream sequence.

Assuming the availability of the HEC-RAS sources, the DDM/HEC-RAS approach could benefit from two levels of parallelism. First, the sparse parallel solver of HEC-RAS performs the sub-unit computation on a multi-core processor. Second, the distribution of the sub-units on a multi-processor cluster may then further improve computing times.

CONCLUSIONS

At a watershed scale, rain-on-grid simulations for the modeling of 2D unsteady flows in response to precipitation input are implemented in the HEC-RAS software by means of a finite volume method, a discretization of the topography, and a sparse parallel linear solver running on a unique processor. Obviously, this may fail due to memory shortage when the watershed of interest and the chosen discretization yield too large linear systems. Such large simulations are good candidates for the design of dedicated partitioning and domain decomposition methods for both the modeling of very large catchments and the reduction of the overall computational time.

Thinking in a load-balanced domain decomposition method, we first propose an area-balanced partition method carried out by considering the flow accumulation information rather than the confluence points and/or the reach length constraint. Second, the domain decomposition method is exemplified for the HEC-RAS software. Special attention has been brought to the set-up of 2D–1D models that combine 2D sub-unit meshes to 1D channels for the extraction of outflow data from upstream sub-units and their propagation to their downstream unit. The workflow is automated by means of the HEC-RAS Controller.

For both case studies, the new partition and decomposition methods have demonstrated a better equilibrium for the sub-basin areas. As a consequence and the computational load that is required for the solution of the related sub-problems, respectively, is balanced between the parts. As expected, the total computational time is substantially reduced when the domain decomposition method is used.

Rain-on-grid simulations over the Saar watershed (1,747 km^2) with a mesh size of 25 m cannot be carried out using HEC-RAS in a straightforward manner because these would yield 2,800,000 mesh elements, overflowing the 16 Gb of RAM memory. The proposed domain decomposition method allows for the solution of such large rain-on-grid simulations, while also reducing the overall computational time. From a hydraulic point of view, resulting discharge and flood maps show very good agreements with public discharge data and a flood extent map produced from SAR Cosmo-Skymed images, respectively.

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest

Uncorrected Proof

ACKNOWLEDGEMENTS

The authors are grateful to H. Yésou (SERTIT) for the communication of the flood extent shapefiles.

DATA AVAILABILITY STATEMENT

All relevant data are included in the paper or its Supplementary Information.

REFERENCES

Apostolopoulos, T. K. & Georgakakos, K. P. 1997 Parallel computation for streamflow prediction with distributed hydrologic models. *Journal of Hydrology* **197**, 1–24.

Bayer, D. M. & Collischonn, W. 2013 *Deforestation Impacts on Discharge of the Ji-Paraná River–Brazilian Amazon. Climate and Land Surface Changes in Hydrology*, Vol. 1, 359th edn. IAHS Publication, Gothenburg, pp. 327–332.

Brunner, G. W. 2010 *HEC-RAS River Analysis System: Hydraulic Reference Manual*. s.l.: U.S. Army Corps of Engineers, Institute for Water Resources, Hydrologic Engineering Center.

Brunner, G. W. 2016 *HEC-RAS River Analysis System, 2D Modeling User's Manual Version 5.0*. U.S. Army Corps of Engineers, Hydrologic Engineering Center, Davis.

David, A. & Schmalz, B. 2021 A systematic analysis of the interaction between rain-on-grid-simulations and spatial resolution in 2D hydrodynamic modeling. *Water* **13**.

Falter, D. *et al.* 2016 Continuous, large-scale simulation model for flood risk assessments: proof-of-concept. *Journal of Flood Risk Management* **9**, 3–21.

Fan, F. M. *et al.* 2016 Flood forecasting on the Tocantins River using ensemble rainfall forecasts and real-time satellite rainfall estimates. *Journal of Flood Risk Management* **9**, 278–288.

Goodell, C. 2014 *Breaking the HEC-RAS Code: A User's Guide to Automating HEC-RAS*. s.l.: H21s.

Goodell, C. 2018 *The HECRAS Controller: A Powerful Feature of HEC-RAS Exposed*. s.l. : s.n.

Jensen, J. L. W. V. 1906 Sur les fonctions convexes et les inégualités entre les valeurs Moyennes. *Acta Mathematica* **30**, 175–193.

Kumar, M. & Duffy, C. J. 2015 Exploring the role of domain partitioning on efficiency of parallel distributed hydrologic model simulations. *Journal of Hydrogeology and Hydrological Engineering* **12**, 2.

Leandro, J., Chen, A. S. & Schumann, A. 2014 A 2D parallel diffusive wave model for floodplain inundation with variable time step (P-DWave). *Journal of Hydrology* **517**, 250–259.

Nash, J. a. S. J. 1970 River flow forecasting through conceptual models part-I: a discussion of principal. *Journal of Hydrology* **10** (3), 282–290.

Pontes, P. R. M. *et al.* 2017 MGB-IPH model for hydrological and hydraulic simulation of large floodplain river systems coupled with open source GIS. *Environmental Modelling & Software* **94**, 1–20.

Rodriguez, F., Andrieu, H. & Morena, F. 2008 A distributed hydrological model for urbanized areas – model development and application to case studies. *Journal of Hydrology* **351**, 268–287.

Schwanghart, W. & Kuhn, N. J. 2010 Topotoolbox: a set of Matlab functions for topographic analysis. *Environmental Modelling & Software* **25**, 770–781.

Schwanghart, W. & Scherler, D. 2014 Topotoolbox 2 – MATLAB-based software for topographic analysis and modeling in Earth surface sciences. *Earth Surface Dynamics* **2**, 1–7.

Singh, J., Knapp, H. V. & Demissie, M. 2005 Hydrological modeling of the Iroquois River. *Journal of American Water Resources Association* **41**, 343–360.

Siqueira, V. A. *et al.* 2016 IPH-Hydro Tools: a GIS coupled tool for watershed topology acquisition in an open-source environment. *RBRH* **21**, 274–287.

Sorribas, M. V. *et al.* 2016 Projections of climate change effects on discharge and inundation in the Amazon basin. *Climatic Change* **136**, 555–570.

Vivoni, E. R. *et al.* 2011 Real-world hydrologic assessment of a fully-distributed hydrological model in a parallel computing environment. *Journal of Hydrology* **409**, 483–496.

Yang, D., Herath, S. & Musiake, K. 1998 Development of a geomorphology-based hydrological model for large catchments. *Proceedings of Hydraulic Engineering* **42**, 169–174.

Zeiger, S. J. & Hubbart, J. A. 2021 Measuring and modeling event-based environmental flows: an assessment of HEC-RAS 2D rain-on-grid simulations. *Journal of Environmental Management* **285**, 112125.

First received 11 June 2021; accepted in revised form 7 January 2022. Available online 22 January 2022

Downloaded from http://iwaponline.com/jh/article-pdf/doi/10.2166/hydro.2022.078/987181/jh2022078.pdf
by guest