water

MDPI

Article

A Mesh Reduced Method for Speeding Up Structured Grid-Based Water Quantity and Quality Models in Large-Scale River Networks

Jin Kang [1,3], Yonggui Wang [2], Jing Xu [2], Shuihua Yang [2] and Haobo Hou [1,*]

[1] School of Resource and Environmental Sciences, Wuhan University, Wuhan 430079, China; kangjin314@163.com

[2] Hubei Key Laboratory of Critical Zone Evolution, School of Earth Sciences, China University of Geosciences, Wuhan 430074, China; wangyg@cug.edu.cn (Y.W.); xujing@cug.edu.cn (J.X.); yangshuihua@cug.edu.cn (S.Y.)

[3] Hubei Provincial Research Institute of Environmental Science, Wuhan 430072, China

* Correspondence: houhb@whu.edu.cn; Tel.: +86-13908644407

Received: 4 February 2019; Accepted: 25 February 2019; Published: 28 February 2019

Abstract: High-precision and efficiently distributed discrete element models for water environment simulation are urgently needed in large-scale river network areas, but most distributed discrete element models are serially computed and need performance improving. Parallel computing and other common methods for models' high performance have large labor costs and are complicated. We put forward a new mesh reduced method for promoting computational efficiency with grid re-organization according to the structure and algorithm characteristics of 2D and 3D numerical models. This simple and cheap method was adapted to a classical three-dimensional hydrodynamic and sediment model (ECOMSED) for model improvement and effective evaluation. Six schemes with different grids were made to investigate the performance of this method in the river network area of the Three Gorges Reservoir Basin. It showed good characteristics of simulation performance and model speed-up. We concluded that the method is viable and efficient for optimizing distributed discrete element models.

Keywords: mesh reduced method; structured grid; discrete element models; large-scale river network; Three Gorges Reservoir Basin

1. Introduction

Distributed discrete element models are important and widely used tools for water environment simulation, such as water quality assessing and forecasting, water conservancy project designing, planning and estimating, etc. [1,2]. Among which, the numerical simulation of water flow and water quality has been widely utilized. Water quantity and quality models have made outstanding contributions to water environment analysis, assessment, and prediction [3–5]. Grid making and numerical discretization are two important and basic technologies in numerical simulation. The finite element method (FEM), the finite difference method (FDM), and the finite volume method (FVM) are typically mature numerical discretization methods and have been applied in fields of computational fluid dynamics, distributed numerical simulation, etc. Grid making is the precondition and the greatest labor-consuming and time-costing work of numerical simulation. It is common that over 60% of the time and manpower are consumed by grid generation in a simulation [6,7]. The quality of grids tremendously affects the accuracy and efficiency of simulations.

Structured grids, unstructured grids, and hybrid grids are three basic types of grids used in numerical simulations. Structured grids are the earliest type of grids, which have a well-aligned structure and explicit adjacency relations. Unstructured grids are more easily created and have better

adaptability to areas with complicated boundaries, while, numerical simulations with structured grids have faster speed, better property of convergence, and more accurate results [8,9]. Structured grids are still the first choice of engineers and researchers for numerical simulations. Many numerical models have adopted structured grids, including the Delft3D [10], the Princeton Ocean Model [11], the ECOMSED [12], and the EFDC [13], just to name a few. In these models, structured grids are identified by a two-dimensional identifier (*i,j*) or three dimensional identifier (*i,j,z*). The simulation zone is determined by *IM* (the number of grids in the *i* direction) and *JM* (the number of grids in the *j* direction). The *IM* × *JM* is used to determine the total amount of grids. Commonly, in the river, the *IM* is set much bigger than the *JM* in the simulation for single-channel, as the river length is much greater than the river width. The basin scale water problems and management requirement desperately need large scale and high precision numerical simulation. Moreover, in the lake, sea, and other river network areas, the *IM* and *JM* may both be set to a huge number, which lead to a greater number of total grids (*IM* × *JM*) and result in low computing efficiency [14]. With *IM* × *JM* for grid numbering, many dry cells will be put into computing. Removing dry cells from the computation has been done by many researchers, such as conformal boundary-fitted mesh generation method [15] and dry cell removing method in TELEMAC-2D [16]. Although, for many existing models, these methods are expensive to adapt for model improvement. At the same time, to overcome speed problems of FEM and FVM models, many model scholars tried parallel computing, block-grid computing, and graphics processing unit (GPU) computing for improving the efficiency of numerical simulation [17–19]. Especially for large-scale water areas, high-performance computational methods are indispensable for the numerical simulation. But it requires heavy work to rewrite codes and build the computing environment during the improvement of existing models with high performance computing. On the other hand, unforeseen mistakes and errors caused by the model improved methods may highly affect the precision of models. For river net, not all of grids in the zone *IM* × *JM* are in wet grids. Setting *IM* × *JM* as computing zones will increase computational cost and prolong computing time compared with that only wet grids been set as computing elements [20]. This drawback has not been reported in prior studies. A straightforward method aiming at improving grid structures and identifying schemes is more appropriate.

To improve the multi-dimensional discrete element models in large scale river network, a mesh reduced method was put forward. In this method, the two dimensional identifier (*i,j*) of structured grids is converted into one dimensional identifier (*ij*); the computing zone *IM* × *JM* was changed into *IJM*; and grids out of water (dry meshes) were removed from the computing zone *IJM*. Unlike high-performance computing methods with a giant workload for manual code improvement, codes could be automatically improved from two dimensional identifiers (*i,j*) to one dimensional identifier (*ij*). The method was implemented to improve the three dimensional hydrodynamic and sediment model named-ECOMSED and tested in the river network of the Three Gorges Reservoir Basin.

2. Model Settings and Framework

2.1. Structured Grids Settings for Models

The finite volume, finite element, and finite difference method are three basic numerical algorithms for computer numerical simulation. Structured and unstructured grids are two primary modes for calculating elements creating of finite difference method. The method, put forward in this paper, is made for structured grid-based models with finite difference method. There are many software platforms, such as the RGFGRID of Delft3D software suite [21], Gambit of Fluent [22], and the mesh tool Turbogird in ANSYS [23] that can generate structured grids, which use the two dimensional identifier (*i,j*) or three dimensional identifier (*i,j,z*) in the Cartesian coordinate system to distinguish grids from one another. Commonly, *i* is the grid number index of grids in the direction along the water flow (*u* direction or ξ_1 direction), *j* is the grid number index of grids in the direction perpendicular to the *i* direction (*v* direction or ξ_2 direction), and *z* is the index of grids in the vertical direction. With

the identifier (*i*,*j*), the spatial topology relationship between grids will be determined conveniently as shown in Figure 1.

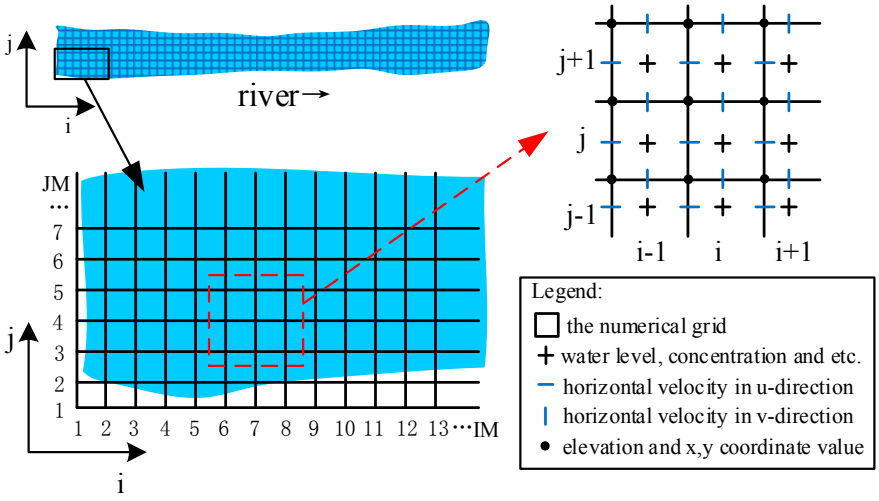

Figure 1. Structured grids identification and calculation in grids.

For non-steady flow and water quality numerical simulations, the water level, flow velocity, pollutants concentration, water temperature, elevation, etc., are defined as grids parameters. Generally, different quantities or parameters are defined and calculated at the different locations in the numerical grid as shown in Figure 1. No matter where the parameters were defined, to calculate grids parameters, the finite differential continuity equation and momentum equation should be used and be solved iteratively. The different dimensional Saint–Venant equations [24] and Navier–Stokes equations [25] are most widely used for flow simulation. To solve these partial differential equations, they should be transformed to the discrete space [26] and solved with different algorithms, such as central difference algorithms, upwind difference algorithms, simple algorithms, etc.

2.2. Problems of Models with Structure Grids Used in River Network Areas

The equations in water environment models are almost solved cyclically using the finite difference method, in which the continuous solution domain will be replaced by finite grid nodes. To ensure discretized equations on every grid solved, at least two-stage looping statement about *i* and *j* should be used to calculate parameters. The structure of the looping statement in *Fortran* is as following:

$$\begin{array}{l} Do\ i{=}1,IM \\ \quad Do\ j{=}1,\ JM \\ \quad\quad \ldots \\ \quad End\ Do \\ End\ Do \end{array} \quad (1)$$

As shown in Equation (1), the computational speed and efficiency of equation solving are dramatically affected by the computing zone $IM \times JM$. In river network areas, the crisscrossed river network makes it a challenge for numerical simulation with finer structured grids. If the main stream and tributaries are long distance, both the *IM* and *JM* should be set as huge number, which leads to a huge $IM \times JM$, as shown in Figure 2. On the other hand, because of the special structure and numbering, a lot of grids covering land areas (dry grids) are useless for modeling (as shown in Figure 2). But in the Equation (1), these dry grids will be put into loop and computation. Some methods set these grids a special sign as not computing grids, but the structure of the loop and model computational efficiency could not been changed.

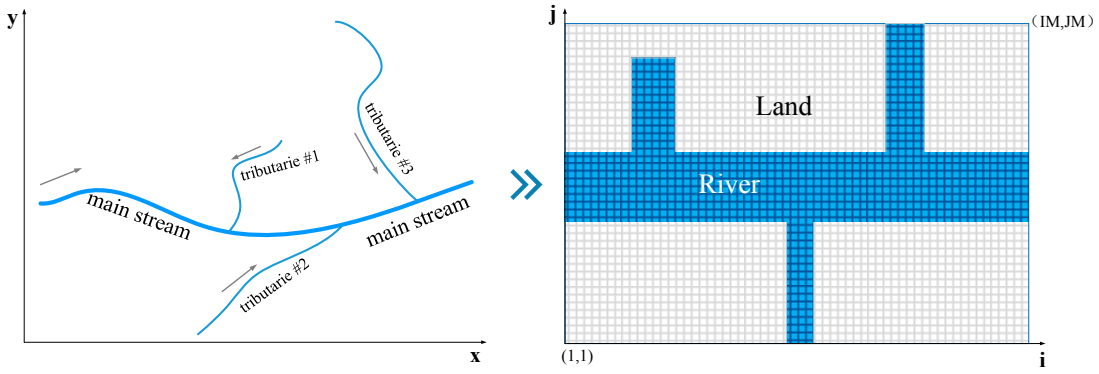

Figure 2. Two dimensional regular grids in river networks. (Grids in grey color are dry grids in the land, grids in blue color are wet grids in rivers).

Generally, the longer the main stream and tributaries, the less proportion of useful grids (i.e., wet grids) in rivers. As useless grids are also calculated in the loop, it takes up a lot of unnecessary computation time and reduces the model's efficiency. Actually, except for the dry useless grids, the number of wet grids in the river network area is not so big, because the tributaries are narrow in a general case.

2.3. Framework of the Mesh Reduced Method

To remove useless grids in the computational domain, a mesh reduced method has been put forward. In this method, the two-dimensional identifier (*i,j*) in the horizontal direction for Cartesian co-ordinate system to distinguish grids from one another is changed to a one-dimensional identifier (*ij*). Principles of the mesh reduced method are: (1) the dry grids should be removed as much as possible; (2) the neighboring relations among grids should not be changed and should be recognizable; (3) the one dimensional grids should be available for moving boundary simulation. According to these principles, the framework of the mesh reduced method contains three steps with three processes, as shown in Figure 3.

Firstly, the boundary of the river network should be extracted. In the river basin, there is an absolute land boundary *(ALB)* over which water can never spill [27]. Comparing the elevation (*zb*) from digital elevation model (DEM) of the river basin with the highest water level (*hel*) on records of all raster points, the *ALB*, where $zb \approx hel$, can be completely made. Sometimes, the river levee can be treated as an ALB.

Secondly, grids generation: According to the ALB, the original grids can be made by the two-dimensional regular grid generating tools. Attribute parameters of grids are set including *i*, *j*, *x*, *y*, and *zb*. During the generation of grids, with the control of ALB, only wet grids can be made, but the identifier of dry grids also exists along with the river network grid drawing and numbering. A two-dimensional mask, *FSM(i,j)*, is used as *FSM(i,j)*=1 at water cells (waterward of the ALB, wet grids, $zb \leq hel$) and *FSM(i,j)*=0 on land cells (landward side of the ALB, virtual dry grids with only *i* and *j*, the other attributes, *x*, *y*, and *zb* are not assigned).

Thirdly, there are three steps during one-dimensional process.

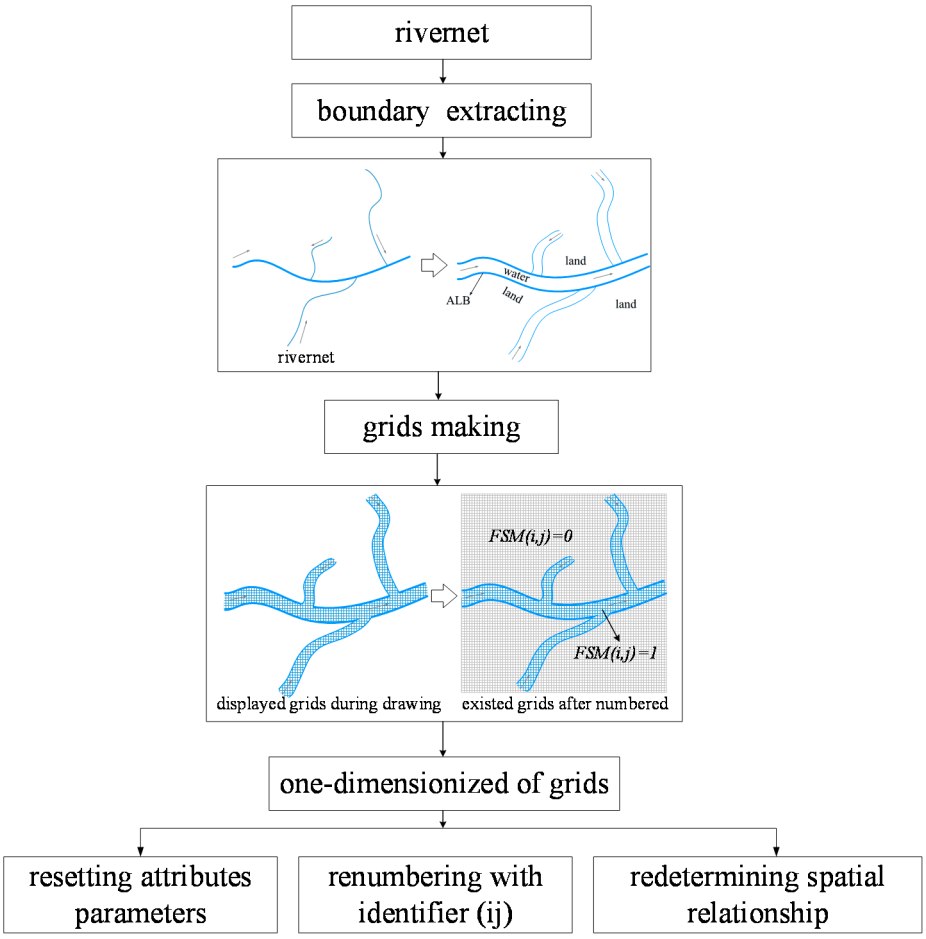

Figure 3. Framework for regular grids one-dimensioned. The ALB is absolute land boundary, over which water can never spill. The FSM is a two-dimensional mask for distinguishing water cells (*FSM(i,j)*=1) and dry cells (*FSM(i,j)*=0).

(1) Attribute parameters re-setup: To make the grids one-dimensioned, the two-dimensional identifier (*i*,*j*) is changed to a one-dimensional identifier (*ij*). The original attributes parameters *x*, *y* and *zb* are retained. To determine the neighboring relationship of grids, the six attributes parameters *i*0, *j*0, *ij*1, *ij*2, *ij*3, and *ij*4 are added. Where, *i*0 is the *i* of the original grid; *j*0 is the *j* of the original grid; *ij*1, *ij*2, *ij*3, *ij*4 are the one-dimensional number ID of neighbored grids in four directions (Figure 4).

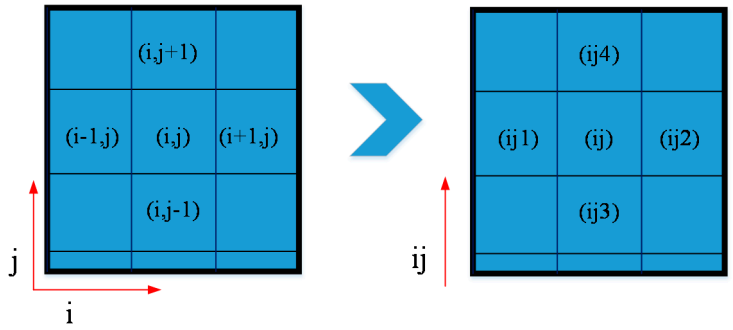

Figure 4. Neighboring relationship between *ij* with *ij*1, *ij*2, *ij*3, and *ij*4.

(2) Grids renumbering with an identifier (*ij*): Set the value *(ij)* of all grids to zero and the amount of all renumbered grids *IJM* to zero. Scan grids from (1,1) to (*IM*,*JM*) in the two-lever looping statement as shown in Equation (1). If the current scanning grid *FSM (i,j)*=1, then *IJM*= *IJM* + 1 and the value (*ij*)

of the grid is set equal to the current *IJM*. The attributes parameters of the grid *IJM* are the same as (*i*,*j*). Renumbering statements codes in *Fortran* are as following.

```
IJM=0
Do i=1,IM
 Do j=1,JM
  IF(FSM(I,j)=0) THEN
     IJM=IJM+1
     x(IJM)=x(i,j)
     y(IJM)=y(i,j)
     zb(IJM)=zb(i,j)
     i0(IJM)=i
     j0(IJM)=j
   End if
 End Do
End Do
```
(2)

(3) Re-determining spatial relationship: In the two-dimensional grids, the number *i* and *j* can be used for neighboring relations judging. As shown in Figure 4, *ij*1 to *ij* is similar to (*i*-1,*j*) to grid *(i,j)*; *ij*2 to ij is similar to (*i*+1,*j*) to grid (*i*,*j*), *ij*3 to *ij* is similar to (*i*,*j*−1) to grid (*i*,*j*) and *ij*4 to *ij* is similar to (*i*,*j*+1) to grid (*i*,*j*).

To obtain the values of *ij*1, *ij*2, *ij*3, and *ij*4, statements codes in *Fortran* can be written as statement Equation (3) after the statement Equation (2).

```
Do ij=1,IJM
  i=i0(ij)
  j=j0(ij)
 Do k=1,IJM
  if(i-1>0)then
   if(i-1.eq.i0(k).and.j.eq.j0(k))then
     ij1(ij)=k
   end if
  end if
  if(i+1<IM)then
   if(i+1.eq.i0(k).and.j.eq.j0(k))then
    ij2(ij)=k
    end if
  end if
   if(j-1>0)then
     if(i.eq.i0(k).and.j-1.eq.j0(k))then
       ij3(ij)=k
    end if
   end if
  if(j+1<JM)then
     if(i.eq.i0(k).and.j+1.eq.j0(k))then
         ij4(ij)=k
     end if
    end if
 End Do
End Do
```
(3)

With this mesh reduced method, the useless dry grids were removed and the amount of grids fells from $IM \times JM$ to IJM.

3. Results and Discussion

3.1. The Improvement of ECOMSED

The ECOMSED is a state-of-the-art three-dimensional hydrodynamic and sediment model, which was developed for shallow water environments—rivers, bays, estuaries—and the coastal ocean, reservoir, and lake simulation. The ECOMSED contains five modules: hydrodynamic module, sediment transport module, wind induced wave module, heat flux module, and particle tracking module, which have functions for water levels, currents, temperature, salinity, tracers, cohesive and non-cohesive sediments, and waves simulation [12]. Recently, the ECOMSED system has been extensively used around the world and proven to be quite robust and reliable over the year [28,29]. The ECOMSED is used with a sigma coordinate system, which is with regular grids in orthogonal Cartesian coordinates in horizontal direction and sigma levels in the vertical direction.

In the grid file named *model_grid* used in ECOMSED and its source codes with *Fortran* language, the i and j are used for expressing the number index of grid element in the ξ_1 and ξ_2 direction; IM and JM are the outer limits of i and j; H is the water depth; FSM is the mask for scalar variables, $FSM(i,j)= 0$ ($H(i,j) \leq 0$) means the grid (i,j) is in the land area; $FSM(i,j)= 1$ $(H(i,j)>0)$ means the grid (i,j) is in the water area.

To improve the ECOMSED, i, j were replaced by ij, and the outer limits of grids were changed to the IJM. On the basis of retaining the original parameters of *model_grid*, $ij1$, $ij2$, $ij3$, and $ij4$ were added into the grid file. Two-level looping statements similar to statement (1) {do i=1,JM; *do* j=1,JM . . . } in the models were replaced by the one-lever looping statement {*do* ij=1, IJM}. Additionally, the ($i-1$,j), ($i+1$,j), etc., were also replaced by $ij1$, $ij2$ and so on, as shown in Table 1. Thus, the model codes were improved by batch replacing (Table 1).

Table 1. Variables and statements are being replaced in models before and after being one-dimensioned (*Fortran* language).

Original Variables	Replaced Variables
do i=1, IM do j=1, JM do j=1, JM or do i=1, IM	do ij=1, IJM
i, j	(ij)
$i-1$, j	$ij1$(ij)
$i+1$, j	$ij2$(ij)
i, $j-1$	$ij3$(ij)
i, $j+1$	$ij4$(ij)
$i-2$, j	$ij1$($ij1$(ij))
$i+2$, j	$ij2$($ij2$(ij))
. . .	. . .

3.2. Application Areas and Model Setup

The Three Gorges Reservoir Basin (TGRB) is the largest reservoir basin in China, which covers an area of 670 square kilometers with hundreds of rivers [30]. The Yangtze River is the main stream of the TGRB, which is 660 km long, and has a 2 km average width from Jiangjin District, Chongqing Municipality to the Three Gorges Reservoir Dam in Yichang City, Hubei Province. There are 26 wide and long tributaries. Recently, along with the development of the Yangtze River Economic Zone and the implementation of the Action Plan for the Prevention and Control of Water Pollution in China, the basin scale accurate simulation of water quantity and quality has been a national demand and urgent need. As part of the Yangtze River (73 km), the Pengxi River (28 km), and Zhuxi River (8 km) in the middle of the TGRB were selected as a river network for the mesh reduced method testing with

ECOMSED, as shown in Figure 5. The model time step placed to 10 seconds and the three modeling schemes with simulation times 1 day, 10 days, and 30 days were made.

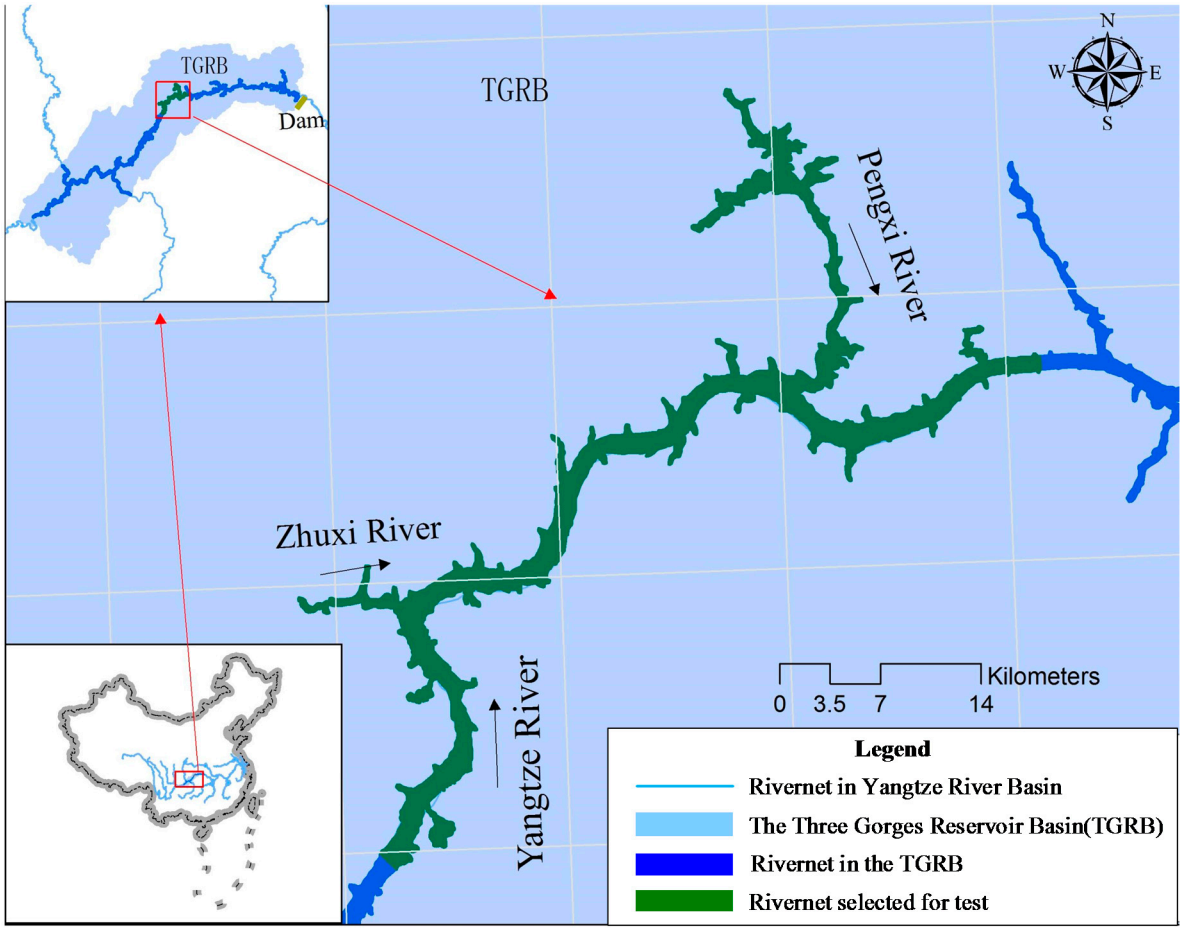

Figure 5. Geographic location and river network selected in the Three Gorges Reservoir Basin.

3.3. Grid Making and Grids One-Dimensioned

The precision of grids is made around 100 meters by Delft3D as shown in Figure 6. In the original two dimensional grids, the number of *IM* is 1064, the number of *JM* is 232; $IM \times JM$ is 246,848.

After grids were one-dimensioned, the total amount of grids (*IJM)* as 26,069. Compared to the original grids, the grid reduction rate was 89.44%. In this case, the reduction rate of grids increased with the selected rivers lengthening.

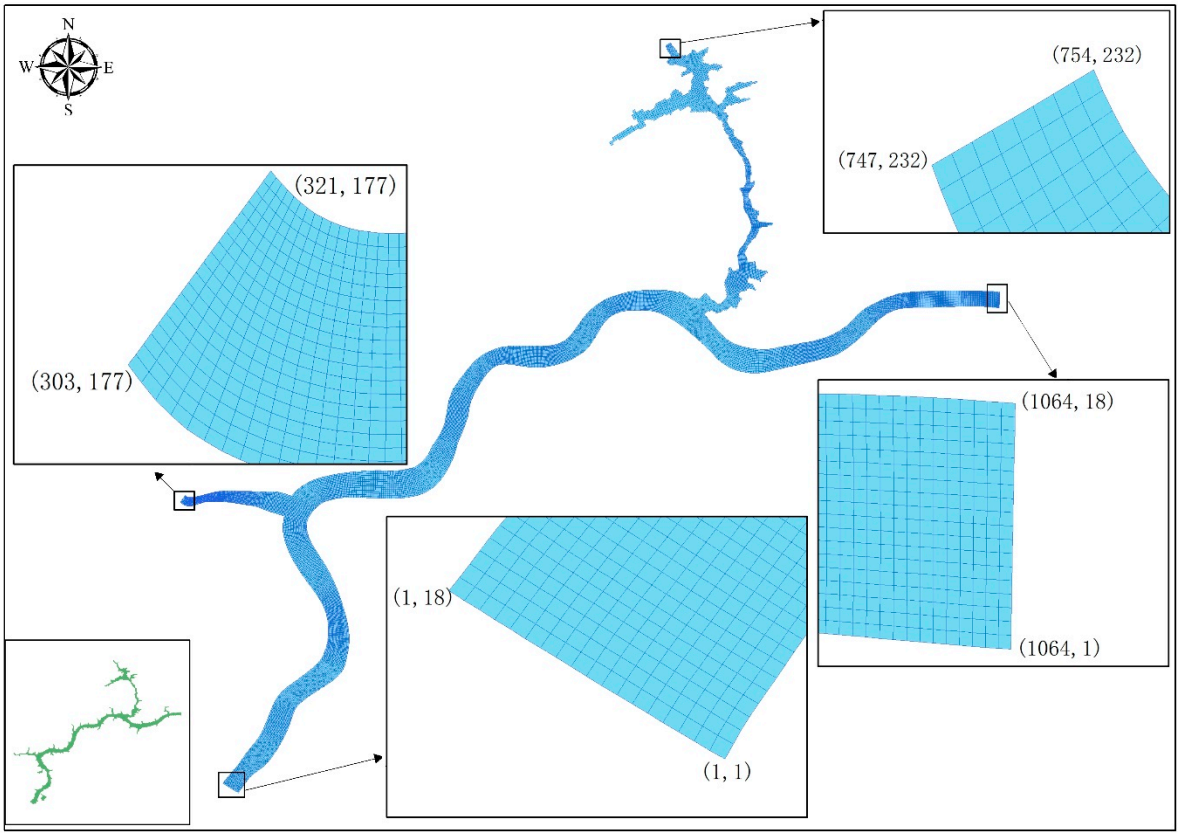

Figure 6. The structure of original two-dimensional grids made by the software Delft3D.

3.4. Accuracy Verification

The relative errors of the results with flow velocity (in both u and v directions), water level, and sediment concentration of between the original ECOMSED model and the one-dimensioned model are shown in Table 2.

Table 2. Relative errors of the results between two models.

Contents	Relative Errors
flow velocity in u direction	0.002%
flow velocity in v direction	0.001%
water level	0.000%
sediment concentration	0.006%

The Table 2 shows that relative errors between the original ECOMSED model and the one-dimensioned model are almost 0%. It is proved that the mesh reduced method will not change the results of the model significantly and will retain the accuracy of the original models.

While, as shown in Table 2, the relative errors were not all equal to zero, which means that, some steps (or factors) in the mesh reduced method will influence the original model. A simple sensitivity analysis was made to assess factors affecting model accuracy. Results showed that the definition of ALB, *IJM* and spatial relationship identifiers (*ij*1, *ij*2, *ij*3, and *ij*4) were sensitivity factors. To avoid the processes of reduced mesh affecting model accuracy, the following principles should be attended.

(1) The girds of ALB should be dry grids, which are close to wet grids.
(2) Grids removed from the original computing zone ($IM \times JM$) should be dry girds, where water can never cover in the real world.
(3) The spatial relationship of all wet girds should be absolutely the same as in the original model.

If these principles are not enforced strictly during grid reduction, the water volume of all grids and the flow field of the river net will be changed. This will bring large and unpredictable errors of one-dimensioned models compared with the original model.

3.5. Improvement of Computational Efficiency

The actual run time of a model is dependent on the number of active grid points, the simulation time, and the time steps. The simulation performance (*sip*) and model speed-up (S_p) are used to evaluate the model efficiency.

The *sip* is defined as:

$$sip = \frac{CPU\ time}{N \cdot IM \cdot JM \cdot L_{max}} \tag{4}$$

where N is the number of steps; IM is the number of grids in i direction; JM is the number of grids in j direction; L_{max} is the number of layers in z direction; the CPU time is time taken for a whole simulation containing the processor time and *I/O* time.

The S_p is defined as:

$$Sp = p_{2d}/p_{1d} \tag{5}$$

where p_{1d} is the CPU time of a model with one-dimensioned grids (1DG), p_{2d} is the CPU time of a model with two-dimensional grids (2DG).

The ECOMSED, with four sigma layers, was used for computational efficiency testing. Based on the original grids, the "derefine tool" in Delft3D was used for grids coarsening. Six schemes with different *IM* and *JM* were made for computational efficiency testing, as shown in Table 3.

Table 3. Six schemes of grids used for computational efficiency testing.

Schemes	IM	JM	IM × JM	Schemes	IJM	Grids Reduction Rate
Scheme 1	1064	232	246,848	Scheme 1′	26,069	89.44%
Scheme 2	532	232	123,424	Scheme 2′	13,115	89.37%
Scheme 3	532	116	61712	Scheme 3′	6492	89.48%

In Scheme 1, the number of grids is 246,848. After being one-dimensioned, 89.44% of grids were reduced. In the Scheme 2 and 3, original grids were defined in the *i* and *j* direction, respectively. The reduction rate of Scheme 2 and 3 were 89.37% and 89.48%. This stated that the mesh reduced method can cut down more than half of the grids. It is intuitively plausible that one-dimensioned grids could double the computational efficiency.

To further assess the optimization effect, the original and improved ECOMSED were used for a 30-day hydrodynamics simulation with six schemes with different numbers of grids. In the 30-day simulation schemes, the central processing unit (CPU) time taken for every 500 steps is shown in Figure 7.

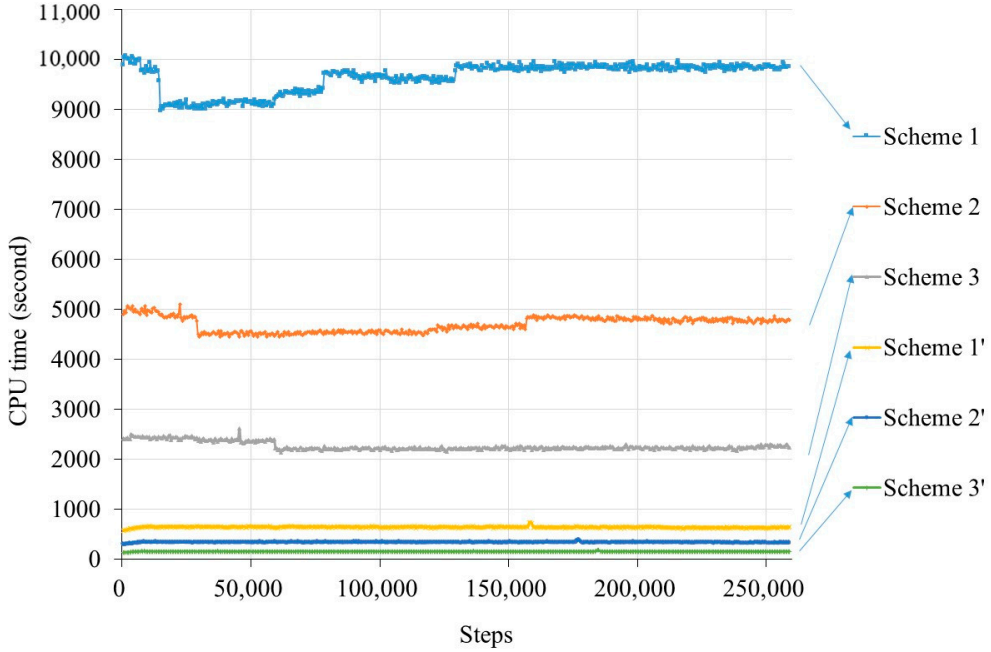

Figure 7. The central processing unit (CPU) time taken for every 500 steps in the 30-day simulation schemes.

With mesh reduced method, the CPU time needed in the computation was tremendously decreased. The CPU time reduction rate increased while the grid was refined. Results of *sip* and *Sp* in the six modeling schemes are shown in Table 4.

Table 4. Results of *sip* and *Sp* in modeling schemes with the three simulation times.

Efficiency Index	Schemes	1 day	10 days	20 days	30 days
Sip (10^5)	Scheme 1	1.11	1.22	1.22	1.22
	Scheme 2	1.13	1.25	1.26	1.26
	Scheme 3	1.03	1.14	1.14	1.14
	Scheme 1′	0.90	0.88	0.93	0.95
	Scheme 2′	0.91	0.87	0.88	0.90
	Scheme 3′	0.87	0.88	0.84	0.83
Sp	Scheme 1′-1	16.26	14.62	14.90	15.09
	Scheme 2′-2	15.91	14.10	14.08	14.23
	Scheme 3′-3	17.05	15.58	15.20	15.07

The simulation performance (*Sip*) is defined as the CPU time per grid point per time step per constituent. The *Sip* of Scheme 1 was $1.1E\times10^5$ on day 1 of the simulation and increased to 1.22×10^5 at 10 days, then maintained. The average reduction rate of *Sip* was 23.17% as shown in Table 5. After being one-dimensioned, values of *Sip* obviously reduced. The *Sip* reduction rate of each scheme distributed between 22.96%~27.15%. The smaller of *Sip*, the better simulation performance and higher efficiency of the model [31]. The average value of *Sip* about Scheme 1 to Scheme 3 was 1.18×10^5. After being one-dimensioned, the average *Sip* of Scheme 1′ to Scheme 3′ decreased to 8.9×10^4 with a reduction rate of 24.6%. Considering the CPU cost, it was considered that the mesh reduced method presented good optimization. For a lot of models, in a similar computing environment, the simulation performance was almost linear and connected with the amount of input/output [32]. Minimizing the output written to the result files, reducing the computing grids, and using a high-speed processor are convenient methods to improve the simulation performance.

Table 5. Reduction rates of *Sip* after being one-dimensioned.

Schemes	1 day	10 days	20 days	30 days	Average
Scheme 1′-1	18.92%	27.87%	23.77%	22.13%	23.17%
Scheme 2′-2	19.47%	30.40%	30.16%	28.57%	27.15%
Scheme 3′-3	15.53%	22.81%	26.32%	27.19%	22.96%

The speed-up is commonly used in the parallel computing for showing the efficiency of high-performance methods [33]. The speed-up ratio of Scheme 1′-1 was 18.92 when the simulation time was 1 day, which was increased to 27.87 and 23.77 with 10 days and 20 days in simulation time. The speed-up ratio of Scheme 2′-2 and Scheme 3′-3 had the same character with Scheme 1′-1. The speed-up ratio of simulation performance will be higher along with the simulation time. Overall, Table 3 presents a speed-up distributing 14.08%~17.05% after grids were one-dimensioned, with an average speed-up of 15.17. The mesh reduced method can definitely speed up the simulation, but the speed-up rate fluctuated with the simulation time going on due to the unstable property of the hardware equipment. The speed-up ratio of the simulation performances also varied with grid number and shape of the river.

In recent years, large-scale environmental problems need highly effective and high-precision environmental assessment and prediction. However, for the established architecture and solution of differential equations, it is a considerable challenge to obtain high-precision results with fast speed in a large-scale river network simulation [34]. In order to increase efficiency of decentralized models, parallel computing methods are widely used. Adopting parallel computing, Kressler et al. [35] obtained a 70% reduction in reconstruction time with 40 processors. Multi-core-CPUs and GPUs are hot research topics for parallel computing. Efremenko et al. [36] achieved a 20–40× speed-up for the multi-stream reverse time migration (RTM) of GPUs. There is no denying that parallel computing could effectually improve the efficiency of finite element models or distributed models [37–39]. But parallel computing is labor intensive, error-prone, and tedious [40]. Compared with other methods for simulation efficiency improving, the mesh reduced method is simple but can get higher efficiency with 14–17× speed-up. Without huge codes for modification and the knowledge of high-performance computing, high speed-up can be achieved by mesh reduced method.

4. Conclusions

For large-scale river network areas, the original two-dimensional structural grids and their numbering method used in many hydrology and water quality models are not cost–benefit algorithms. Before adapting parallel computing methods, the structure and grids used in these models can be optimized. We put forward a mesh reduced method for structured grid-based multi-dimensional model optimization. In order to characterize the scheme, numerical tests were performed on a 3D hydrodynamic model with structured grids and finite difference method. The model modified was applied to a river network with three rivers of the Three Gorges Reservoir Basin (TGRB). Six schemes with different number of grids before and after being one-dimensioned were put into test. The method cut down on a large amount of grids—over 89%. During the simulation of 30 days, results indicated that the mesh reduced method exhibited higher simulation performance and better speed-up during a large-scale river network hydrodynamic simulation. The method is simple and cheap (in manpower cost and CPU cost), and efficient if implemented wisely.

To achieve high-precision and high-efficiency models, many fruitful methods have been proposed. With one-dimensioned models, many structured grid-based water quantity and quality models can be improved; however, the one-dimensioned method is only helpful for structured grids models. With the one-dimensioned method, the organization pattern of structured grids will be changed, but there is no substantial optimization of numerical models. This means that models after being one-dimensioned are still serially computed. If adopted in extremely complicated or great wide-range river-net areas, these models still need efficiency improvement with other methods. Parallel computing is the most widely

adopted method for model computational efficiency improvement. After being one-dimensioned, adapting parallel computing methods for modelling high-performance improvement is the next significant work.

Author Contributions: J.K. designed the project and drafted the manuscript. Y.W. provided writing ideas and supervised the study. H.H. finalized the manuscript. S.Y. and J.X. collected and calculated the data. All authors reviewed the manuscript.

Acknowledgments: The authors gratefully thank the support by the National Natural Science Foundation of China (41807471).

Conflicts of Interest: The authors declare no conflict of interest in any aspect of the data collection, analysis or the preparation of this paper.

References

1. Mattiussi, C. An Analysis of Finite Volume, Finite Element, and Finite Difference Methods Using Some Concepts from Algebraic Topology. *J. Comput. Phys.* **1997**, *133*, 289–309. [CrossRef]
2. Bu, W.; Tang, Y.; Wu, Y.; Yang, J. Finite difference/finite element method for two-dimensional space and time fractional Bloch–Torrey equations. *J. Comput. Phys.* **2014**, *293*, 264–279. [CrossRef]
3. Chen, W.B.; Liu, W.C.; Hsu, M.H. Water quality modeling in a tidal estuarine system using a three-dimensional model. *Environ. Eng. Sci.* **2011**, *28*, 443–459. [CrossRef]
4. Rathnayaka, K.; Malano, H.; Maheepala, S.; George, B.; Nawarathna, B.; Arora, M.; Roberts, P. Seasonal demand dynamics of residential water end-uses. *Water* **2015**, *7*, 202–216. [CrossRef]
5. Mohajeri, S.H.; Righetti, M.; Wharton, G.; Romano, G.P. On the structure of turbulent gravel bed flow: Implications for sediment transport. *Adv. Water Resour.* **2016**, *92*, 90–104. [CrossRef]
6. Fischer, C.; Nana, G.; Selberherr, S. Finite difference, boundary-fitted grid generation for arbitrarily shaped two-dimensional simulation areas. *Comput. Methods Appl. Mech. Eng.* **1993**, *110*, 17–24. [CrossRef]
7. Velho, P.; Schnorr, L.M.; Casanova, H.; Legrand, A. On the validity of flow-level tcp network models for grid and cloud simulations. *Acm Trans. Model. Comput. Simul.* **2013**, *23*, 1–26. [CrossRef]
8. Jorgenson, P.C.E.; Pletcher, R.H. An implicit numerical scheme for the simulation of internal viscous flow on unstructured grids. *Comput. Fluids* **1996**, *25*, 447–466. [CrossRef]
9. Hou, J.; Simons, F.; Mahgoub, M.; Hinkelmann, R. A robust well-balanced model on unstructured grids for shallow water flows with wetting and drying over complex topography. *Comput. Methods Appl. Mech. Eng.* **2013**, *257*, 126–149. [CrossRef]
10. And, C.H.K.W.; Govardhan, R. Vortex-induced vibrations. *Annu. Mech.* **2004**, *36*, 413–455.
11. Ezer, T.; Mellor, G.L. Sensitivity studies with the north atlantic sigma coordinate princeton ocean model. *Dyn. Atmos. Ocean.* **2000**, *32*, 185–208. [CrossRef]
12. Blumberg, A.F. *A Primer for ECOMSED User Manual: Versionl*; Hydroqual. Inc.: Mahwah, NJ, USA, 2002; pp. 1–188.
13. Irvine, K.N.; Pettibone, G.W. Dynamics of indicator bacteria populations in sediment and river water near a combined sewer outfall. *Environ. Technol. Lett.* **1993**, *14*, 531–542. [CrossRef]
14. Quirk, J.J. An alternative to unstructured grids for computing gas dynamic flows around arbitrarily complex two-dimensional bodies. *Comput. Fluids* **1994**, *23*, 125–142. [CrossRef]
15. Lin, B.; Chandler-Wilde, S.N. A depth-tntegrated 2d coastal and estuarine model with conformal boundary-fitted mesh generation. *Int. J. Numer. Methods Fluids* **2015**, *23*, 819–846. [CrossRef]
16. Medeiros, S.C.; Hagen, S.C.; Weishampel, J.F. Comparison of floodplain surface roughness parameters derived from land cover data and field measurements. *J. Hydrol.* **2012**, *452–453*, 139–149. [CrossRef]
17. Cao, W.; Xu, C.F.; Wang, Z.H.; Yao, L.; Liu, H.Y. CPU/GPU computing for a multi-block structured grid based high-order flow solver on a large heterogeneous system. *Clust. Comput.* **2014**, *17*, 255–270. [CrossRef]
18. Navarro, C.A.; Hitschfeldkahler, N.; Mateu, L.A. A Survey on Parallel computing and its applications in data-parallel problems using GPU architectures. *Commun. Comput. Phys.* **2014**, *15*, 285–329. [CrossRef]
19. Xu, S.; Huang, X.; Zhang, Y.; Oey, L.Y.; Xu, F.; Fu, H.; Yang, G. gpuPOM: A GPU-based princeton ocean model. *Geosci. Model Dev. Discuss.* **2014**, *7*, 7651–7691. [CrossRef]
20. Medeiros, S.C.; Hagen, S.C. Review of wetting and drying algorithms for numerical tidal flow models. *Int. J. Numer. Methods Fluids* **2013**, *71*, 473–487. [CrossRef]

21. Schmalz, R. ROMS high resolution hindcasts for delaware river and bay. *Am. Soc. Civ. Eng.* **2014**, 67–88.
22. Srinivas, G.; Gowda, B.P.M. Aerodynamic Performance Comparison of Airfoils by Varying Angle of Attack Using Fluent and Gambit. *Appl. Mech. Mater.* **2014**, *592–594*, 1889–1896. [CrossRef]
23. Kong, X.; Wu, D.J.; Cai, C.S.; Liu, Y.Q. New strategy of substructure method to model long-span hybrid cable-stayed bridges under vehicle-induced vibration. *Eng. Struct.* **2012**, *34*, 421–435. [CrossRef]
24. Litrico, X.; Fromion, V.; Baume, J.P.; Arranja, C.; Rijo, M. Experimental validation of a methodology to control irrigation canals based on Saint-Venant equations. *Control Eng. Pract.* **2005**, *13*, 1425–1437. [CrossRef]
25. Equation, N.S. Navier-Stokes equations. *Acta Appl. Math.* **1989**, *28*, 295–296.
26. Aoki, K.I.; Sato, D. Solving the QCD non-perturbative flow equation as a partial differential equation and its application to dynamical chiral symmetry breaking. *Prog. Theor. Exp. Phys.* **2013**, *4*, 5834–5838. [CrossRef]
27. Larsen, S.E. *The Atmospheric Boundary Layer over Land and Sea: Focus on the Offshore Southern Baltic and Southern North Sea Region*; DTU Wind Energy: Roskilde, Denmark, 2013; pp. 1–36.
28. Yuan, S.; Lin, L.; Amini, F.; Tang, H. Numerical Study of Turbulence and Erosion of an HPTRM-Strengthened Levee under Combined Storm Surge Overflow and Wave Overtopping. *J. Coast. Res.* **2014**, *30*, 142–157.
29. Wang, Y.G.; Yang, Y.Q.; Chen, X.L.; Engel, B.A.; Zhang, W.S. The moving confluence route technology with wad scheme for 3D hydrodynamic simulation in high altitude inland waters. *J. Hydrol.* **2018**, *559*, 411–427. [CrossRef]
30. Shi, Y.Y.; Xu, G.H.; Wang, Y.G.; Engel, B.A.; Peng, H.; Zhang, W.S.; Cheng, M.L.; Dai, M.L. Modelling hydrology and water quality processes in the Pengxi River basin of the Three Gorges Reservoir using the soil and water assessment tool. *Agric. Water Manag.* **2017**, *182*, 24–38. [CrossRef]
31. Kaazempur-Mofrad, M.R.; Ethire, C.R. An efficient characteristic galerkin scheme for the advection equation in 3-D. *Comput. Methods Appl. Mech. Eng.* **2002**, *191*, 5345–5363. [CrossRef]
32. Allcock, B.; Bester, J.; Bresnahan, J.; Chervenak, A.L.; Foster, I.; Kesselman, C.; Tuecke, S. Data management and transfer in high-performance computational grid environments. *Parallel Comput.* **2002**, *28*, 749–771. [CrossRef]
33. Itoh, T.; Yamaguchi, A.; Kyotani, T.; Hanaoka, T.A.; Mizukami, F. High-performance bio-sensor with enzymes immobilized on mesoporous membranes: Nanosized pores just corresponding to the size of an enzyme improve the stability of the sensor drastically. *Adv. Porous Mater.* **2016**, *4*, 157–165. [CrossRef]
34. Zounmevo, J.A.; Afsahi, A. A fast and resource-conscious MPI message queue mechanism for large-scale jobs. *Future Gener. Comput. Syst.* **2014**, *30*, 265–290. [CrossRef]
35. Kressler, B.; Spincemaille, P.; Prince, M.R.; Yi, W. Reduction of reconstruction time for time-resolved spiral 3D contrast-enhanced magnetic resonance angiography using parallel computing. *Magn. Reson. Med.* **2006**, *56*, 704–708. [CrossRef] [PubMed]
36. Efremenko, D.S.; Loyola, D.G.; Doicu, A.; Spurr, R.J.D. Multi-core-CPU and GPU-accelerated radiative transfer models based on the discrete ordinate method. *Comput. Phys. Commun.* **2014**, *185*, 3079–3089. [CrossRef]
37. Johan, Z.; Mathur, K.K.; Johnsson, S.L.; Hughes, T.J.R. An efficient communications strategy for finite element methods on the Connection Machine CM-5 system. *Comput. Methods Appl. Mech. Eng.* **1994**, *113*, 363–387. [CrossRef]
38. Kennedy, G.J.; Martins, J.R.R.A. A parallel finite-element framework for large-scale gradient-based design optimization of high-performance structures. *Finite Elem. Anal. Des.* **2014**, *87*, 56–73. [CrossRef]
39. Pettit, J.R.; Waiker, A.E.; Lowe, M.J.S. Improved detection of rough defects for ultrasonic NDE inspections based on finite element modeling of elastic wave scattering. *AIP Conf. Proc.* **2014**, *1581*, 521–528.
40. Chen, C.H.; Chu, W.C.; Lu, C.W.; Chung, Y.C.; Yang, D.L. A Parallel Program Generation Environment for Solving PDEs on Distributed Memory Computing Environments. *Asian J. Inf. Technol.* **2012**, *11*, 1025–1038.