Environmental Modelling & Software 101 (2018) 51–63

Contents lists available at ScienceDirect

Environmental Modelling & Software

journal homepage: www.elsevier.com/locate/envsoft

Parallelization of a distributed ecohydrological model

Ning Liu [a, b, *, 1], Mohsin Ahmed Shaikh [c, 1], Jatin Kala [a], Richard J. Harper [a, b], Bernard Dell [a, b], Shirong Liu [b], Ge Sun [d]

[a] School of Veterinary and Life Sciences, Murdoch University, South Street, Murdoch, WA 6150, Australia
[b] Chinese Academy of Forestry, Beijing 10091, China
[c] Pawsey Supercomputing Centre, Perth, WA, Australia
[d] U.S. Department of Agriculture, Forest Service, Southern Research Station, Eastern Forest Environmental Threat Assessment Center, Raleigh, NC 27606, USA

ARTICLE INFO

Article history:
Received 17 January 2017
Received in revised form
21 November 2017
Accepted 22 November 2017
Available online 4 January 2018

Keywords:
High performance computing
Ecohydrological modeling
Distributed memory parallelism
Shared memory parallelism
Water and carbon fluxes

ABSTRACT

WaSSI-C is an ecohydrological model which couples water and carbon cycles with water use efficiency (WUE) derived from global eddy flux observations. However, a significant limitation of the WaSSI-C model is that it only runs serially. High resolution simulations at a large scale are therefore computationally expensive and cause a run-time memory burden. Using distributed (MPI) and shared (OpenMP) memory parallelism techniques, we revised the original model as dWaSSI-C. We showed that using MPI was effective in reducing the computational run-time and memory use. Two experiments were carried out to simulate water and carbon fluxes over the Australian continent to test the sensitivity of the parallelized model to input data-sets of different spatial resolutions, as well as to WUE parameters for different vegetation types. These simulations were completed within minutes using dWaSSI-C, whereas they would not have been possible with the serial version. The dWaSSI-C model was able to simulate the seasonal dynamics of gross ecosystem productivity (GEP) reasonably well when compared to observations at four eddy flux sites. Sensitivity analysis showed that simulated GEP was more sensitive to WUE during the summer compared to winter in Australia, and woody savannas and grasslands showed higher sensitivity than evergreen broadleaf forests and shrublands. Although our results are model-specific, the parallelization approach can be adopted in other similar ecosystem models for large scale applications.

Software availability

Name of software: dWaSSI-C
Developer: WaSSI-C was developed by Ge Sun; dWaSSI-C was developed by Ning Liu and Mohsin Shaikh
Contact address: Ge Sun (gesun@fs.fed.us); Ning Liu (N.Liu@murdoch.edu.au)
Cost: Free
Software availability: Contact the developers (Github)

1. Introduction

Ecohydrological models describe the interactions between water, vegetation and climate by coupling multiple hydrological and ecological processes. They are very useful tools in assisting land-managers and policy-makers to simulate 'what-if' scenarios, such as the effects of projected climate change on water resources and the terrestrial carbon cycle. Process-based distributed ecohydrological models are commonly used by ecohydrologists, as they not only achieve higher accuracy than empirical models (Chen et al., 2015), but also couple the terrestrial water, energy and biogeochemical cycles. Additionally, process-based models allow for investigations over much larger spatial and temporal resolutions as compared to traditional field studies (Fatichi et al., 2016).

Remote sensing data are increasingly used as inputs to ecohydrological models to achieve more accurate simulations in comparison with models based solely on mathematical theory (Liu et al., 1997; Cao and Woodward, 1998; Sun et al., 2011b). However, without careful evaluation and quality control, high resolution remote sensing data could induce systematic biases in ecohydrological simulations (Zhao et al., 2005). Eddy covariance flux towers, which provide continuous measurements of ecosystem level fluxes of water and carbon (Baldocchi et al., 2001), have been used for

* Corresponding author. School of Veterinary and Life Sciences, Murdoch University, South Street, Murdoch, WA 6150, Australia.
E-mail address: N.Liu@murdoch.edu.au (N. Liu).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.envsoft.2017.11.033
1364-8152/

evaluating remote sensing data and ecohydrology models (Stoy et al., 2006; Sjostrom et al., 2013; Raczka et al., 2013; Zhou et al., 2014). As the eddy covariance technique simultaneously measures water and carbon fluxes, it reflects interactions between water and vegetation at the ecosystem scale, and is therefore widely used to analyze changes in ecosystem carbon fluxes (Xiao et al., 2012) and water use efficiency (WUE) (Xiao et al., 2013).

The Water Supply Stress Index and Carbon model (WaSSI-C), developed by Sun et al. (2011b), is an example of ecohydrological models which uses both remote sensing and eddy flux observations to simulate the coupling of water and carbon fluxes. The accuracy of the WaSSI-C model has been evaluated at a monthly time-scale at 72 United States Geological Survey (USGS) gauging stations (Sun et al., 2015) and in the upper Zagunao watershed, a sub-catchment of the MinJiang watershed, in China (Liu et al., 2013a, b). Comparisons of the simulated streamflow against observations at the 72 USGS stations showed good overall model performance, with correlation coefficients ranging from 0.71 to 1.0 (Sun et al., 2015). With state-of-the-art remote sensing technologies, ecohydrological models, such as WaSSI-C driven with remote sensing data, can simulate water and carbon processes at a very high resolution globally. Such high temporal and spatial resolution simulations are needed to investigate the interactions between the water and carbon cycles and for ecosystem management, both at small and large scales (Wood et al., 2011).

State-of-the-art remote sensing sensors have spatial resolutions ranging from centimeters on unmanned aerial vehicle platforms, to meters on satellite platforms, which provide a great opportunity for understanding vegetation dynamics. Application of such data would have immediate benefits. For example, hyper-resolution modeling at 100 m or finer resolutions would allow for much better representation of the effects of spatial heterogeneity of topography, soils and vegetation cover on hydrological dynamics (Wood et al., 2011). This in turn will allow for a better representation of processes that are poorly represented in the current generation of models, such as slope and aspect effects on surface incoming and reflected solar radiation, and consequent effects on snowmelt, soil moisture redistribution and evapotranspiration. However, processing hyper-resolution modeling implies large data input into the resident memory of a Central Processing Unit (CPU) followed by computation and consequent writing of the results to disk. Thus, data movement is an obvious bottleneck in this process. Additionally, advanced computational algorithms are needed to process satellite and other datasets via computationally demanding data assimilation procedures. Making use of multiple cores on a processor and distributing the computational domain to map onto multiple processors can be a prospective scalable solution to these issues.

The models of Kollet et al. (2010) and Le et al. (2015) perform each simulation unit independently, and hence the model computational domain can easily be broken down into smaller individual segments as there is no dynamic interaction. However, for models including dynamic interaction with inter-connected simulation units, a dynamic parallelization method is required. In order to parallelize a distributed model with inter-connected simulation units, Li et al. (2011) developed a dynamic parallelization method to balance computation load, resulting in higher speedup and efficiency of parallel computing. Zhang et al. (2016) further revised this dynamic parallelization method for hydrological model calibration using high-performance computing (HPC) systems. Nonetheless, at present, massively parallel computational methods are not often implemented within ecohydrological models. With the increasing availability of high resolution remote sensing products as well as high performance computers, there is an opportunity to improve the accuracy of ecohydrological models and reduce their computing time.

The WasSSI-C model is a useful ecohydrological model, but a major limitation is that the model does not operate in parallel. Continental-scale simulations carried out at a pixel scale require large computational resources to generate output and post-processing. Therefore, the aim of this study was to develop a distributed version of the model (dWaSSI-C), capable of using high spatial and temporal resolution remote sensing input data. Continental scale simulations were carried out over Australia to test the sensitivity of water and carbon estimates to key model parameters. Specifically, the sensitivity of gross ecosystem production (GEP) to WUE was investigated for each vegetation type, using the new parallelized model.

2. Materials and methods

2.1. WaSSI-C model

The WaSSI-C model is an ecohydrological model developed by Sun et al. (2011b) and typically used to simulate monthly fluxes. The main purpose of WaSSI-C is to couple the water and carbon cycles with WUE (a ratio of GEP to Evapotranspiration (ET)) which is derived from global eddy flux observations. WaSSI-C consists of two empirical sub-models - a water supply stress index model and an empirical carbon model. The input data include vegetation type, soil parameters, monthly mean meteorological forcing and remote sensed leaf area index (LAI); and the main outputs are runoff, ET, ecosystem respiration (ER) and GEP.

Potential evapotranspiration (PET) is derived from LAI, precipitation (P) and reference evapotranspiration (ET_0) (Sun et al., 2011a). In order to consider the effect of actual soil water storage on water fluxes, ET is calculated using the Sacramento soil moisture accounting (SAC-SMA) model (Anderson et al., 2006) with PET. The Carbon sub-model is an eddy-flux derived WUE empirical carbon model, which calculates carbon fluxes from ET and WUE. A schematic of the WaSSI-C model is illustrated in Fig. 1 and a more detailed description of the WaSSI-C model can be found on the model's website (http://www.forestthreats.org/research/tools/WaSSI).

2.1.1. Monthly mean meteorological forcing

To simulate carbon and water fluxes over the Australian continent, monthly gridded rainfall and temperature data were obtained from the Ecosystem Modeling and Scaling Infrastructure (eMAST) ANUClimate v1.0 dataset at 0.01° spatial resolution from 1970 to 2013 (Hutchinson, 2014). The daily rainfall data (*in situ*) used to generate the gridded product are from the Australian Bureau of Meteorology (BoM) network of weather stations. ANUClimate integrates a new approach to interpolate the station data to a regular grid using an improved background anomaly interpolation method and a new “proximity to the coast” modifier (Hutchinson, 2014).

2.1.2. Monthly mean leaf area index (LAI)

The 0.01° gridded monthly LAI for this study were generated by the Land-atmosphere interaction group at Beijing Normal University (http://globalchange.bnu.edu.cn/research/lai/) (Yuan et al., 2011). These data were originally obtained from Moderate resolution imaging spectroradiometer (MODIS) LAI products, and then were improved by a two-step integrated method. In the first step, missing and poor quality data were optimized with a modified temporal spatial filter and MODIS LAI's quality control information. This database was then fitted by TIMESAT (a software package for analysing time-series of satellite sensor data) using the Savitzky-Golay smoothing model to further reduce the potential noise (Jonsson and Eklundh, 2004). The maximum rather than the mean

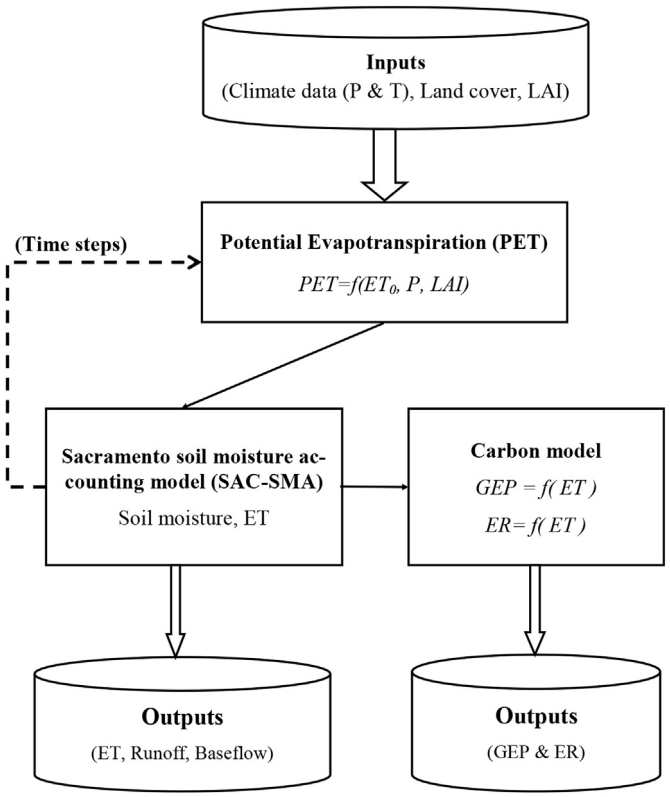

Fig. 1. Schematic diagram of the WaSSI-C model. T, P, LAI, ET, ET_0, GEP and ER are temperature, precipitation, leaf area index, evapotranspiration, reference evapotranspiration, gross ecosystem productivity and ecosystem respiration, respectively. The core of the WaSSI-C model runs continuously for each time step (month).

of 8-day's LAI in each month was used as the monthly LAI value to omit the effect of potential noise and clouds (Zhao et al., 2005).

2.1.3. Static soil and vegetation data

Gridded soil properties at 0.01° resolution were derived from the soil hydrological properties for the Australia dataset (McKenzie et al., 2000). This dataset provides Australia's soil hydrological properties for A and B horizons, which are derived from soil mapping based on the Atlas of Australia Soils. The key soil hydrological properties used by WaSSI-C include the soil depth, soil plant available water holding capacity, soil thickness, saturated hydraulic conductivity, field capacity, wilting point and plant available water holding capacity. Using these soil properties, 11 soil parameters for both A and B horizons were developed using Anderson's method (Anderson et al., 2006). These 11 parameters are required by the SAC-SMA model to calculate the ET in WaSSI-C.

Vegetation types were derived from the MODIS land-cover product (MCD12Q1) which has a resolution of 500 m (https://lpdaac.usgs.gov/dataset_discovery/modis/modis_products_table/mcd12q1). This dataset uses the International Geosphere-Biosphere Programme (IGBP) classification scheme which consists of 17 general land-cover types, including 11 natural vegetation classes and 6 other land classes (Friedl et al., 2010; Taylor et al., 2012). In this study, the original 500 m MODIS land cover data were interpolated to 50, 20, 10, 5 and 1 km using the "Majority" resample algorithm (dominant vegetation type) (http://pro.arcgis.com/en/pro-app/tool-reference/data-management/resample.htm) for running simulations at different resolutions in order to test the performance of parallelization method to different resolutions (described in more detail in Section 2.3).

2.2. Framework of dWaSSI-C

The basic computing unit of the original WaSSI-C model is a discrete watershed (Sun et al., 2011b). By using averaged climatic variables, remote sensing and other land surface properties for each watershed, the original WaSSI-C model provides watershed scale water and carbon estimates. However, each watershed generally consists of various vegetation types and soil properties, therefore a study at the watershed scale cannot provide enough spatial information on water and carbon processes. To resolve this issue, the parallelized and distributed version of WaSSI-C, referred to as dWaSSI-C from here onwards, was developed to simulate processes at the pixel scale.

The choice of pixel resolution is one of the key considerations when running the model. This is illustrated in Fig. 2 showing the distribution of vegetation types across Australia at 0.5*0.5 km resolution versus 50*50 km resolution. At coarser resolutions, areas close to the coastlines are more likely to be missclassified as compared to higher resolutions, which illustrates the need for hyper-resolution modeling. However, a key limitation is that the input data size for the WaSSI-C model increases exponentially with the increase in resolution of gridded input data as shown in Fig. 3. For example, the input dataset size increased from 50 MB to 850 GB when the pixel size was reduced from 50*50 km to 0.5*0.5 km, while the number of grid points increased from 5644 to 53,971,840. The increasing input data size makes this a memory-bound problem, which can be handled by splitting the input data into smaller chunks to fit into the available main memory of a computing node.

2.2.1. Computing infrastructure

Simulations were executed on the Pawsey Supercomputing Centre's computers called *Magnus* and *Zeus*. *Magnus* is a Cray XC40 series supercomputer using Intel Xeon E5-2690 v3 "Haswell" processors (2.6 GHz), with a total of 1488 nodes with 24 CPUs per node providing a total of 35,712 processor cores. Each compute node has access to 64 GB of memory. *Zeus* is an SGI cluster with compute nodes having Intel Xeon E5-E5-2670 v2 "Ivybridge" processor (2.5 GHz), with 20 cores, and RAM ranging from 128 to 512 GB per node. *Magnus* is designed for parallel applications whereas *Zeus* is designed for high memory serial applications. To identify the bottlenecks in the code, serial simulations at different resolutions were carried out on *Zeus* so as to make use of the larger amount of memory (described in more detail in the next sub-section). Based on these results, parallel simulations were then carried out using *Magnus*, to take advantage of the higher number of compute nodes (described in more detail in section 2.3).

2.2.2. Identifying the bottlenecks

Before parallelizing a model, it is important to first identify the bottlenecks or hotspots where most of the compute time is spent in sequential execution of the code. This is illustrated in Table 1 showing the breakdown of time spent in different activities by the compute resource while running a dWaSSI-C simulation at 5 km resolution on a single core on *Zeus*. Table 1 shows that most of the time spent by dWaSSI-C was executing CPU instructions. Most of the time in CPU was spent fetching data from main memory upon encountering cache misses. Improving data structures could improve this metric but the code is clearly memory bandwidth limited.

Table 2 shows the variation in runtime memory usage with increasing resolution, and illustrates an inherent limitation of the model with respect to how large the input datasets can be ingested when running the model on a single core. The memory requirement for coarser to finer scales (50–0.5 km) grows exponentially. Scaling up the memory by adding more RAM on the

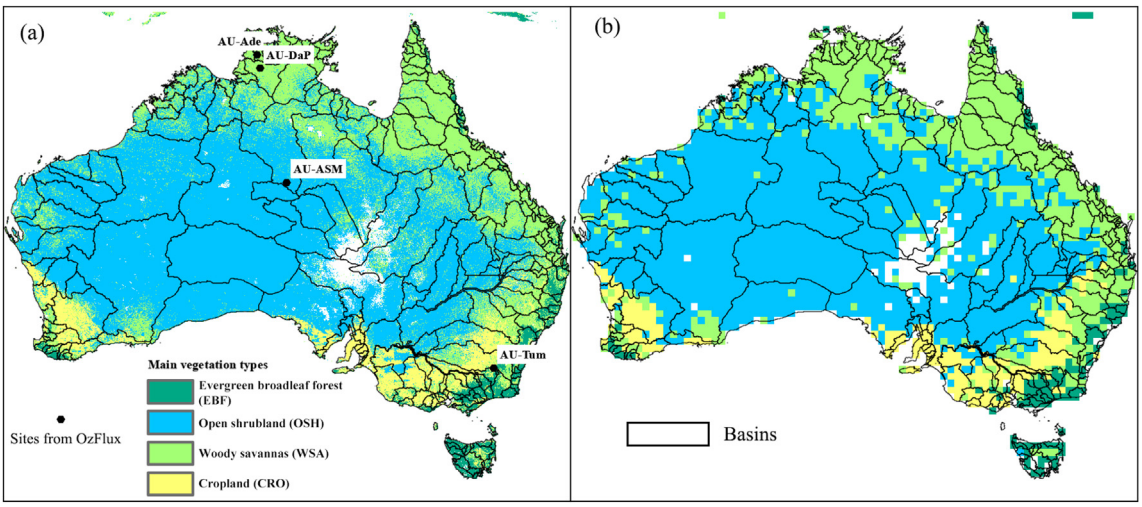

Fig. 2. Distribution of vegetation types (MCD12Q1) across the Australian continent at (a) 0.5*0.5 km and (b) 50*50 km resolution. The dots on Panel (a) shows the locations of the eddy flux sites used for model evaluation.

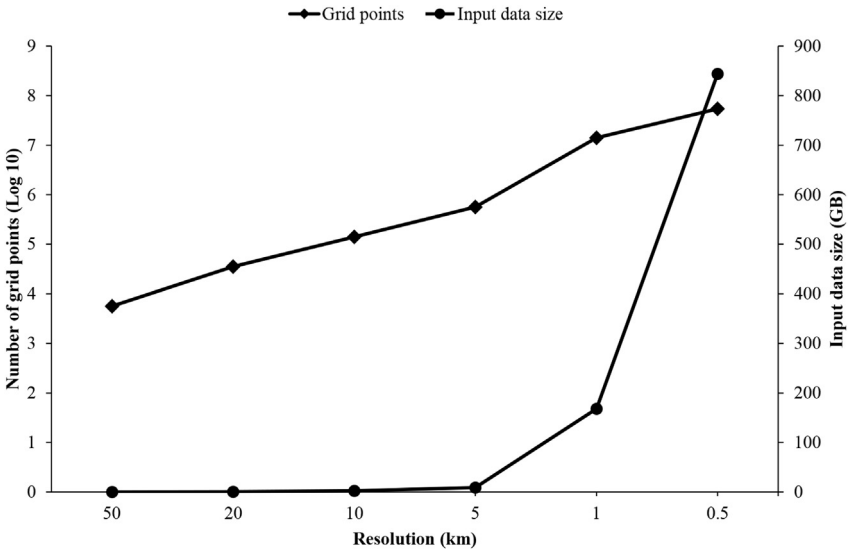

Fig. 3. Number of grid points and input data size as a function of simulation resolution.

Table 1
Runtime profiling of single threaded simulations at 10 km and 5 km resolution with dWaSSI-C model operated on the *Zeus* cluster.

Profiling processes			Resolution	
			10 km	5 km
Total time to solution			633.4 s	783.7 s
	CPU Instructions		**96%**	**95%**
		Compute	14.5%	16.5%
		Memory access	77.2%	75%
		Branch prediction fails	8.3%	8.5%
	File I/O		**4%**	**5%**
Runtime memory usage			29.4 GB	41.6 GB

same node may not be a viable option since the code is memory bandwidth limited. Additionally, such large shared memory machines are expensive and thus rare. Scaling out, using distributed memory architecture, is a plausible solution as it provides two benefits, a) runtime memory can be extended to many nodes, and b) the bandwidth limitation can be hidden by optimizing the chunk of the grid running on each node.

Table 3 breaks down the time spent in executing the CPU instructions in Table 1 which is 95% for the 5 km resolution simulation. Almost all of the time spent by the CPU is in the subroutine *waterbal*, which spends 45% of the time computing instructions in the local scope of the subroutine and the 38% in calling other

Table 2
Runtime memory usage of single-threaded dWaSSI-C simulations at 20, 10, 5 and 1 km resolution on the *Zeus* cluster. The 1 km resolution simulation failed due to insufficient memory and hence the last reported metric is presented.

Resolution	Runtime Memory Usage
20 km	8.74 GB
10 km	29.40 GB
5 km	41.60 GB
1 km	>431.26 GB

subroutines. Most of the instructions are simple arithmetic operations but the amount of memory access calls by the CPU from Table 1 suggests that the data locality could be improved. Data locality/cache reuse (or lack of it) is a systemic problem in the code and it would therefore be advantageous to investigate the use of more CPUs. We decomposed the grid into subgrids/subdomains and allow either threads or independent processes to processes these subgrids/subdomains.

2.2.3. Model parallelization

The basic computing unit of dWaSSI-C is the pixel and there is no interaction between pixels. This implies that each pixel can be computed/processed independently. For each pixel on the 2D modeling domain, the input and output data can be seen as an n-dimensional array. The first two dimensions of this n-dimensional array are the timestamps which include the year and the month. As the simulation for each pixel is temporally successive, domain decomposition is conducted to reduce computational time. Thus the domain containing all the pixels can be split into N subdomains. Two means of parallelism, a shared memory and a distributed memory model, were investigated to explore their respective advantages and limitations.

To exploit the multi-core architecture of modern day chips, a shared memory model was implemented using OpenMP, which is an application programming interface for implementing multi-platform shared memory programming models. The model splits the domain into N subdomains and distributes the work to OpenMP threads. The advantage of using this model is that it is able to utilize all the cores on a multi-core chip. All the threads can access the global memory address space in a thread-safe manner. The intermediate results are buffered and written by the master thread at the end. It is important to highlight the inherent limitation of the shared memory model that the maximum number threads are limited by the physical core count of the processor and how many logical CPUs each core can present. The shared memory approach can have scalability issues when it comes to codes with frequent memory access, as is the case here.

To address the inherent limitations of OpenMP, i.e. the shared memory approach, a distributed memory model was implemented as an "embarrassingly parallel" model using Message Passing Interface (MPI), leveraging the independent computing for each pixel in this study. In parallel computing, an embarrassingly parallel workload is one where the problem domain can be decomposed into subdomains which can then run independently on compute units with minimal need of synchronization by communicating between them. The gridded domain was segmented in N subdomains and distributed to p MPI-processes. All MPI-processes get a subdomain of equal size if the total girds N are exactly divisible by the total number of MPI-processes p. If not, then the last MPI-process gets the remainder subdomain additionally. Unlike OpenMP, each MPI process has its local memory address space and can only communicate to another MPI process via an MPI library call. When these subdomains are distributed to an MPI-process, each MPI-process may use OpenMP threads and employ the shared memory model, as discussed above, as a second level of parallelism. Thus, the shared memory model using OpenMP can either be used as standalone parallel model if running on a single multicore node or as a hybrid model with MPI, as illustrated in Fig. 4 showing the framework parallelizing dWaSSI.

2.2.4. I/O optimization

Reading and writing large datasets to the disk must keep up to avoid I/O becoming a bottleneck. In the case of shared memory parallelism where the dWaSSI-C runs on a single shared memory node with multiple OpenMP threads, file I/O is handled by a single thread. This is because disk I/O is not a thread safe operation. Failing to scale I/O performance with the improving compute performance would make the code I/O bound (this is discussed in more detail in Section 3.1).

In the case of the distributed memory model, one MPI-process does all the I/O, which means scattering and gathering input and output respectively using MPI. This adds a communication cost and also creates a limitation that the data should fit into the memory of MPI-process doing the file I/O. The other possibility could be that each MPI-process does its own file I/O. This is a good strategy at small scales but high performance filesystems e.g. Luster, favor either shared parallel files or the case where a subset of MPI-processes are responsible to do the file I/O. The later case is more scalable but requires some communication to scatter and gather data onto the designated MPI-processes. In our distributed memory implementation of dWaSSI-C, each MPI process reads and writes to a shared parallel file using the MPI-IO application programming interface. Additionally, to reduce the archiving requirements, the output was stored in binary format as opposed to ASCII.

2.3. Model simulations in parallel

To show-case the usefulness of the parallelized dWaSSI-Cmodel, sensitivity tests were carried out to test the sensitivity of GEP to the WUE parameter values used by the model. This parameter was

Table 3
Breakdown of CPU time spent executing various instructions. Only the most expensive subroutines are listed.

Subroutine	Self %	Child %	Instruction/Subroutine
wassicbzb	0	100	PROGRAM WaSSICBZB
waterbal	44.6	37.8	CALL WATERBAL(ICELL, …, MNDAY) …
	8.9	0	RUNLAND(I,J,M,DAY) = SURFRO + PBF + SBF + INF
	8.5	0	GEPLAND(I,J,M,DAY) = GEP(J,M)
	7.5	0	ETLAND(I,J,M,DAY) = ET(J,M)
	2.7	0	(UZTWM(I)+LZTWM(I)))
	2.2	0	ELSEIF (LADUSE(I).EQ.8 ….)THEN
	2.1	0	ETUZTW(J,M) = ET(J,M)*(UZTWC/UZTWM(I))
	1.8	0	220 IF(UZTWC …) GO TO 225
			113 others called by waterbal

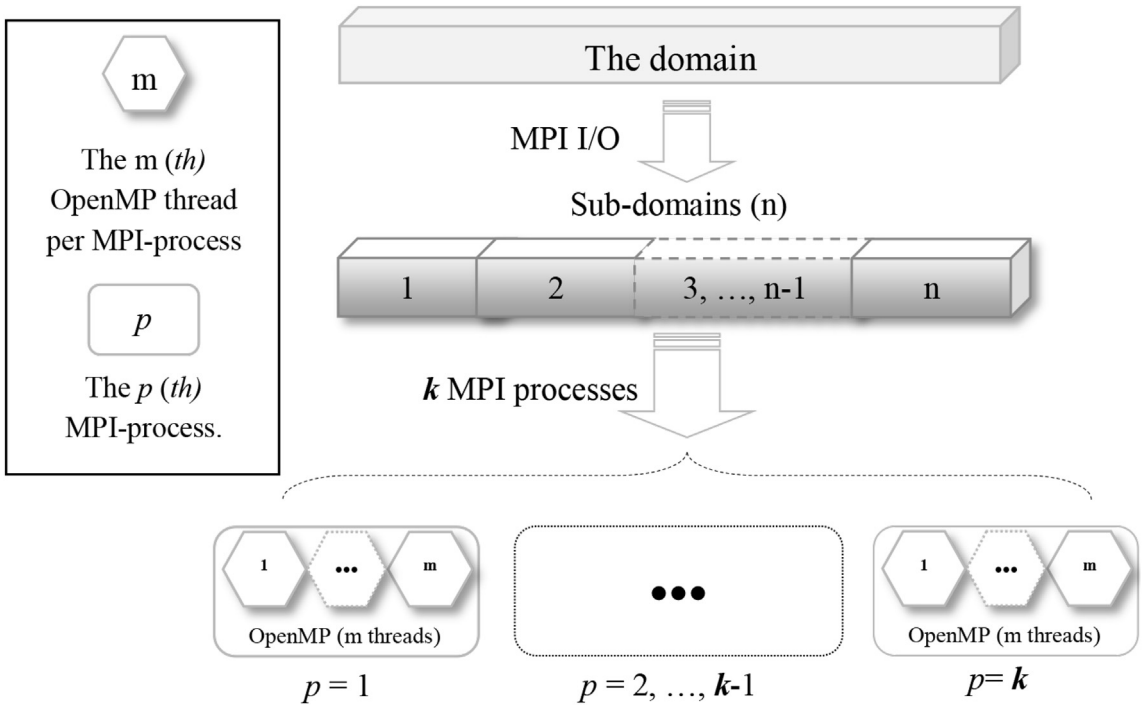

Fig. 4. Schematic diagram of the parallelization methods for the dWaSSI model (the core of WaSSI-C is shown in Fig. 1). The domain can be divided into n sub-domains and evenly passed to *p* Message Passing Interface (MPI) tasks. With the hybrid approach, there can be m (m ≥ 1) OpenMP threads for each MPI task.

chosen as it couples water and carbon processes and is a key parameter for estimating GEP. Sensitivity tests were carried out by running simulations with WUE by ± 1 * SD for each vegetation type as shown in Table 4. In order to evaluate the model, 4 eddy flux sites from the OzFlux network (Beringer et al., 2016; Trudinger et al., 2016) were used (http://data.ozflux.org.au/). The vegetation types for these 4 sites (AU-ASM (Cleverly, 2011), AU-Tum (van Gorsel, 2013), AU-Ade (Beringer, 2013a) and AU-DaP (Beringer, 2013b)) are evergreen needleleaf forest (ENF), evergreen broadleaf forest (EBF), woody savannas (WSA) and grassland (GRA), respectively, and the locations for these sites are shown in Fig. 2a.

3. Results

3.1. Shared memory model using OpenMP

Fig. 5 demonstrates the speed up achieved by introducing shared memory parallelism using OpenMP. The profiling was run on single large memory node on Pawsey's *Zeus* cluster, with one thread per core, and up to 16 cores, for simulations at 10 and 5 km resolution, respectively. OpenMP was implemented to parallelize the loops processing the input data and in both cases the OpenMP parallelization did not yield promising results, with a reduction factor of approximately 1.2 when using 16 OpenMP threads versus a single thread. This slight improvement in the overall compute time was largely due to the fraction of the compute time spent in executing the *nested DO loops* parallelized by OpenMP.

In order to better understand the relatively poor performance of implementing OpenMP, the total compute time was sub-divided into the amount of time spent carrying out OpenMP operations, referred to as the *omp region*, the amount of time in carrying out *File I/O* (i.e. time spend in reading and writing to files), and finally, the amount of time spent in executing the subroutine *output*, which formats the outputs and actually writes output files to disk. This is illustrated in Fig. 6, which shows that the amount of time spent in *omp region* quickly plateaued after 8 threads for both the 10 km and 5 km resolution simulations. The majority of the time was spent in executing the subroutine *output*, which runs sequentially as File I/O

Table 4
The water use efficiency (WUE) parameter and its standard deviation (SD) for vegetation types used by the WaSSI-C model (Sun et al., 2011b).

Vegetation type	WUE Parameter	SD	R^2
Cropland (CRO)	3.13	1.69	0.78
Evergreen broadleaf forest (EBF)	2.59	0.54	0.92
Grassland (GRA)	2.12	1.66	0.84
Open shrubland (OSH)	1.33	0.47	0.85
Woody Savannas (WSA)	1.26	0.77	0.80

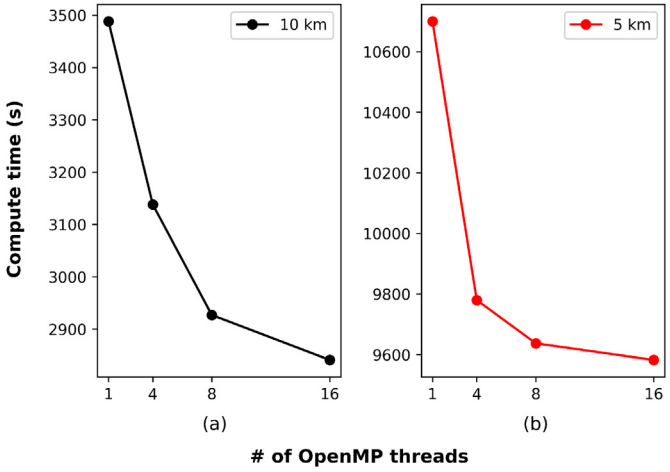

Fig. 5. Compute time (s) as a function of the number of OpenMP threads for dWaSSI-C simulations at (a) 10 km resolution and (b) 5 km resolution.

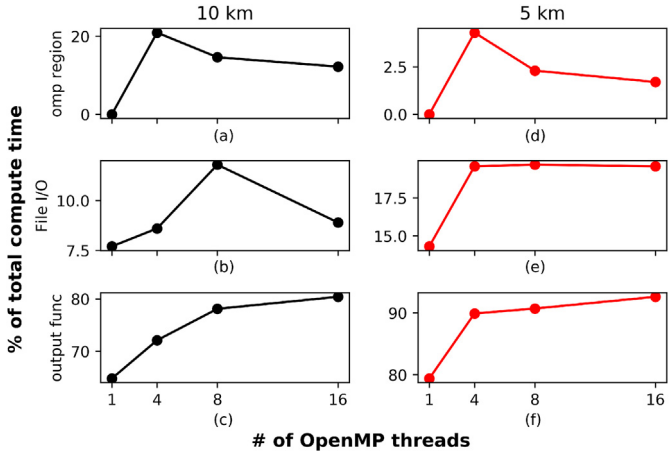

Fig. 6. Percentage of total compute time shown in Fig. 5 for (a-c) 10 km and (d-f) 5 km resolution dWaSSI-C simulations. *omp region* refers to the time spent in OpenMP calls. *output func* is the time spent in executing the *output* subroutine which includes preparing the data for writing i.e. formatting and the write statements in the code. The time spent in *File I/O* including the write statements in the *output* subroutine is shown in (b) and (e). The *output* subroutine in OpenMP version of dWaSSI-C code is executed outside the OpenMP region as file I/O is not thread safe.

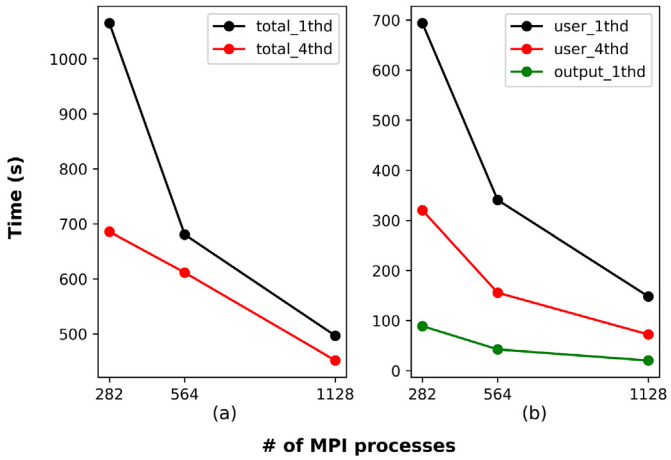

Fig. 7. Variation in compute time with increasing MPI processes, shown in (a), with 1 and 4 OpenMP threads for dWaSSI-C simulations at 0.5 km resolution. (b) shows the time spent in executing user subroutines and also isolates the time spent in executing the subroutine *output* in particular.

is thread unsafe and therefore sits outside the scope of OpenMP. This subroutine takes approximately 65% of the overall compute time at 10 km resolution and approximately 82% at 5 km resolution with a single OpenMP thread, and this increases rapidly with increasing numbers of threads. Thus, introducing OpenMP made the code I/O bound. However, File I/O of Fig. 6 suggests that a mere 8-20% of total time in either case of input datasets is spent in actual file I/O. Upon closer inspection, it was found that most of the time was being spent in Fortran's *format* calls within the *output* subroutine. Thus, this operation became extremely expensive as the compute time in loops was reduced by introducing OpenMP.

3.2. Distributed memory model using MPI and hybrid MPI/OpenMP implementations

As has been described earlier, the distributed memory model uses MPI to decompose and distribute the subdomains to worker MPI processes. If only a single OpenMP thread is used, then the implementation is MPI-only, and if more than 1 OpenMP threads are used, then the implementation is a hybrid MPI/OpenMP model as the worker MPI process can then spawn additional OpenMP threads and thus introduce shared memory parallelism. Fig. 7 (a) shows the reduction in compute time by using an increasing number of MPI processes with 1 OpenMP thread (i.e. a pure MPI implementation), and 4 OpenMP threads (i.e. a hybrid implementation) for a 0.5 km resolution dWaSSI-C simulation. The compute time is reduced by approximately a factor of 2 by using more MPI processes when using a single OpenMP thread. The effectiveness of using 4 rather than 1 OpenMP threads decreases as the number of MPI processes increases, which suggests that using an increasing number of OpenMP threads may not be efficient for very high resolution simulations. To better understand where compute time is spent, Fig. 7 (b) shows a break down of the time spent by the *output* subroutine and what we refer to as *user* calls, which includes all other calls, including the main compute intensive subroutine *waterbal* (Table 3). The use of more MPI processes clearly results in a notable reduction in the amount of time spent in *user* calls, but notably, the compute time taken by the *output* subroutine also decreases, albeit not as efficiently as *user* subroutines. Hence, the introduction of MPI-IO to read and write files in parallel has removed I/O bottleneck identified in the previous section.

Since the use of 4 versus 1 OpenMP thread leads to a marked reduction in compute time when using 282 MPI processes (Fig. 7 (a)), we further investigated the impact of using more OpenMP threads as shown in Fig. 8. Using 8 OpenMP threads only results in a slight improvement compared to 4, and using 16 actually results in a slight increase in compute time compared to 8 threads. Most of the improvement occurs when calling the *user* subroutines, *waterbal* in particular, whereas the *output* subroutine remains relatively unaffected as there is no OpenMP region in it. The total compute time does not improve beyond 4 OpenMP threads, as the OpenMP overhead (denoted as *omp_ovhd* in Fig. 8(d)), i.e. the time cost imposed by the operating system in managing threads becomes larger than the time spent in the OpenMP region of the code (denoted as *omp_region* in Fig. 8(d)). In summary, for high resolution simulations using dWaSSI-C, the MPI approach should be adopted, i.e. scaling to more MPI nodes, while minimizing the number of OpenMP threads if a hybrid approach is used.

To further improve on the file writing performance, the Lustre

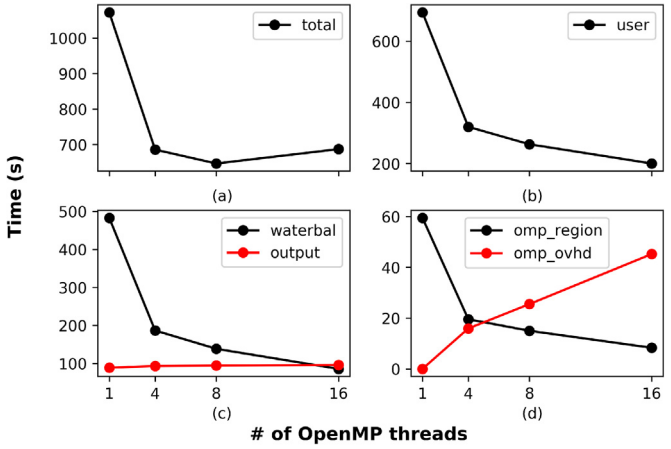

Fig. 8. (a): Variation in total compute time for a dWaSSI-C simulation at 0.5 km resolution using 282 MPI processes, with increasing number of OpenMP threads, (b): Compute time spent in executing *user* subroutines which is *total - operating system* time, (c): Time spent in the *waterbal* and *output* subroutines and (d): the time spent in the OpenMP region of the code (*omp_region*) versus the corresponding overhead imposed by the operating system to manage threads respectively (*omp_ovhd*).

filesystem file striping feature was explored. File striping is a feature of Lustre filesystem which is at the core of it being a high performance parallel filesystem. A file can be written on more than one disk or Object Storage Targets (OSTs). On a shared filesystem, repeated access to data residing on a single disk can pose a bottleneck. This is because data from other users may also be located on the same disk thus serializing the I/O operation. Striping the file on multiple storage object reduces the footprint of the data on a single OST and the I/O requests are fulfilled quickly. Multiple clients can read or write on different OSTs at the same time. Fig. 9 shows the effect of using an increasing number of stripe counts on compute time for a 0.5 km resolution dWaSSI-C simulation. Using 16 stripe counts significantly reduced the compute time, but improvements beyond 64 stripe counts were minimal. The results shown in Figs. 7 and 8, corresponded to the input datasets being read from 16 OSTs and the output files written on 112 OSTs.

In summary, the key factor in improving the performance of dWaSSI-C is to scale to more MPI processes. In the following section, we implement this approach to test the sensitivity of the model to one of its key parameters.

3.3. Sensitivity of GEP to WUE of dWaSSI-C

Using the distributed version of dWaSSI-C, 5 km resolution simulations from 2000 to 2013 were conducted to investigate the sensitivity of the model to the WUE parameter (Table 4), and the simulated GEP was compared to fluxnet observations at 4 OzFlux sites (Fig. 2). This is illustrated in Fig. 10 showing monthly time series of the simulated and observed GEP at the 4 OzFlux sites. The dWaSSI-C model is able to capture the seasonal variations in GEP as compared to the observed GEP well. Simulated time series of GEP were strongly correlated to observed GEP ($p < 0.05$). For the sensitivity analysis, the monthly mean simulated GEP by the dWaSSI-C model was compared to observations at each of the 4 OzFlux sites as illustrated in Fig. 11 showing the monthly climatological comparisons of GEP between the dWaSSI-C model with WUE $\pm$ 1 * SD and observations at the 4 OzFlux sites. Fig. 12 shows the spatial sensitivity of GEP to WUE during winter and summer. For those four vegetation types, EBF showed the lowest sensitivity (< 2.5 g C m^{-2} d^{-1}) to variations in the WUE parameter, while GEP of WSA at the AU-Ade site and GRA at the AU-DaP site demonstrated very high sensitivity to WUE, especially during the growing season (about 10 g C m^{-2} d^{-1}) (Figs. 11 and 12). In addition, Fig. 11 illustrates that there is a higher variation of GEP in summer than in winter at all of the 4 sites.

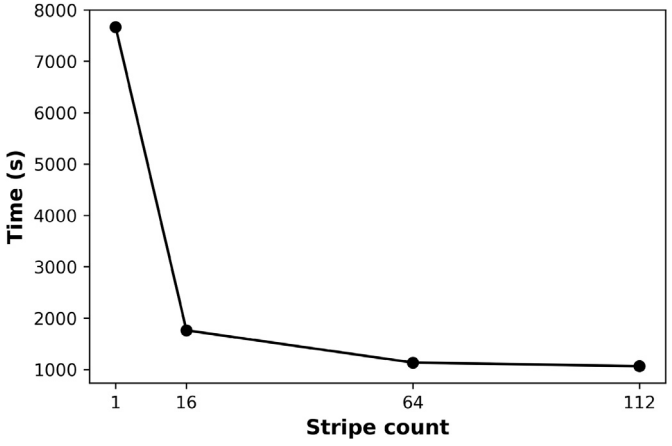

Fig. 9. Effect of Lustre stripe count on the compute time for dWaSSI-C simulations at 5 km resolution.

Spatially, Fig. 12 demonstrates that GEP in coastal areas was more sensitive to WUE than inland areas. Moreover, GEP of grassland in north Australia had the highest seasonal variation and was strongly sensitive to WUE, especially in summer, which is the monsoon season. The absolute difference in GEP between control and control +1 * SD for WSA was less than 1 g C m^{-2} d^{-1} in winter but more than 8 g C m^{-2} d^{-1} in summer. The GEP of OSH in inland areas was not sensitive to WUE irrespective of the season. GEP of EBF and CRO along the south coast areas demonstrated moderate sensitivity to WUE, with absolute differences between the control and control +1 * SD of approximately 2.5 g C m^{-2} d^{-1}.

The root mean square errors of the regressions between simulated and observed GEP time series at the AU-ASM, AU-Tum, AU-Ade and AU-DaP were 0.8, 1.5, 1.9 and 2.1 g C m^{-2} d^{-1}, respectively. With the default WUE for each vegetation type, GEP at AU-DaP showed the best agreement between dWaSSI-C and OzFlux observations. The dWaSSI-C model showed a tendency to underestimate GEP at the AU-Tum site in summer, but overestimated GEP in summer at the AU-Ade and AU-ASM sites (Fig. 11).

3.4. Water and carbon estimates over the Australian continent from 2000 to 2013

Water and carbon fluxes over the Australian continent were simulated using the parallelized dWaSSI-C model at a 5*5 km using the default parameters from 2000 to 2013. The mean annual ET and GEP were 0.54 mm d^{-1} and 1.0 g C m^{-2} d^{-1}, respectively, and the annual mean ET generally showed a similar spatial pattern to GEP as shown in Fig. 13. There were explicit gradients of water and carbon fluxes from inland arid zones to coastal temperate and tropical zones. Specifically, ET ranged from approximately 0.4 to 1.95 mm d^{-1} from the middle eastern and western inland arid zones to the eastern and southwestern temperate and northern tropical coastal zones, while GEP ranged from approximately 0.7 to 4.4 g C m^{-2} d^{-1}.

As for vegetation types, EBF, which was mainly distributed along the southwest and southeast temperate zones, showed the highest carbon productivity with mean annual ET and GEP of 1.9 mm d^{-1} and 4.9 g C m^{-2} d^{-1}, respectively. Although EBF makes up only approximately 1% of total vegetation cover, this vegetation type still plays an important role in carbon sequestration, as it is photosynthetically active throughout the year. The dominant vegetation in Australia is OSH, which accounts for 67% mainly in the inland arid zone, and showed the lowest carbon productivity, with ET and GEP around 0.7 mm year^{-1} and 1.6 g C m^{-2} d^{-1}, respectively. This was mainly the result of the shortage of water. CRO and WSA, which were located in the transitional areas between forest and shrublands and account for approximately 10% of vegetation cover, showed a similar capacity for carbon sequestration (3.1 g C m^{-2} d^{-1}), but the ET of WSA (1.7 mm d^{-1}) was much higher than that of CRO (1.2 mm year^{-1}). SA, in the northern tropical zone, had modest carbon sequestration capacity, with ET and GEP of approximately 2.1 mm d^{-1} and 4.6 g C m^{-2} d^{-1}, respectively.

Overall, the dWaSSI-C model can capture the seasonal cycles of both water and carbon fluxes reasonably well. However, biases between observations and simulated GEP can still be large (Fig. 10) and therefore, the model needs further calibration and evaluation before answering more specific scientific questions, such as the impacts of future climate change on the water and carbon cycles over the Australian continent. Since the main aim of this paper is to show-case the usefulness of parallelizing the model, further model calibration and evaluation is outside the scope of this paper, but will be the subject of future work.

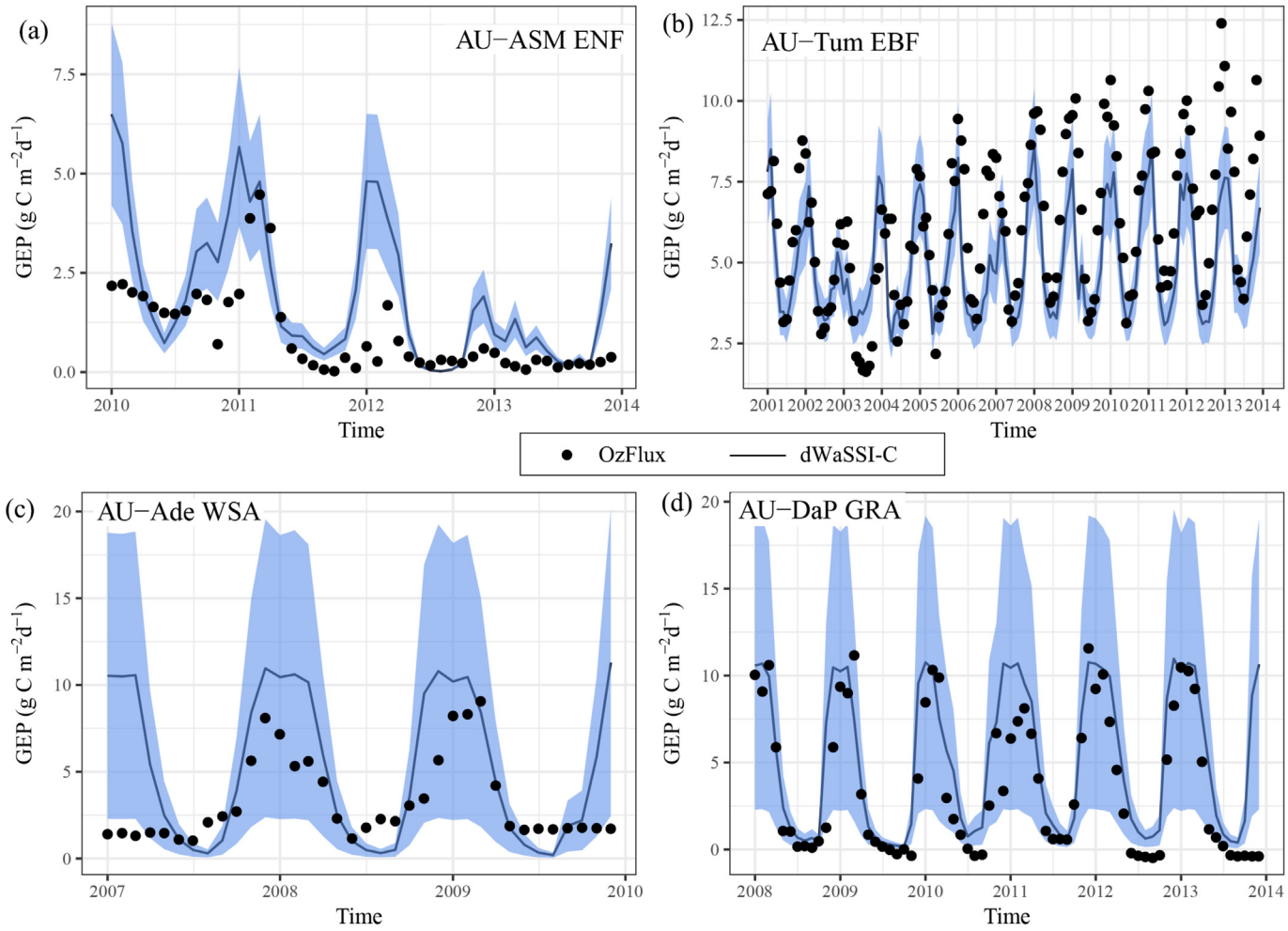

Fig. 10. Time series of simulated by dWaSSI-C (lines) and observed by OzFlux (points) gross ecosystem productivity (GEP, g C m^{-2} d^{-1}) at (a) AU-ASM, (b) AU-Tum, (c) AU-Ade and (d) AU-DaP. The shaded region represents GEP simulated by the dWaSSI-C model with WUE ± 1 * SD. ENF, EBF, WSA and GRA represent evergreen needleleaf forest, evergreen broadleaf forest, woody savannas and grassland, respectively.

4. Discussion and conclusions

4.1. Parallel model

The dWaSSI-C model is memory-bound and using shared-memory parallelism (OpenMP) alone, has significant limitations (Fig. 5). The use of distributed memory parallelism (MPI) allows for the model domain to be decomposed into smaller chunks, which are processed independently and this allows the code to run much more efficiently (Fig. 7, single thread use). Using the hybrid approach with up to 16 OpenMP threads only did not increase performance markedly (Fig. 8), and hence, running the model as MPI-only is also a viable option. Ideally, the computation time should decrease linearly with the increase in MPI processors. In the real world however, adding more resources does not necessarily make the code run faster. This is because the time taken by the parallelized computational part of the code reduces as the problem domain is decomposed and distributed more finely. Thus, at one point, the overhead of communication and/or disk I/O surpasses the expense of computation. The scaling graphs in Fig. 7 demonstrate that there is an optimum number of MPI processes to maximize the usage of the computation resource of a computation node. This optimum number or sweet spot is dependent on the size/resolution of the problem domain, and hence, scaling tests should be carried out for different resolutions prior to running ensembles. Another advantage of the using the distributed memory model is the scalability in terms of memory. A problem size posed by, for example, the 0.5 km resolution simulation, is impractically large to simulate on any high-end desktop or even a shared memory machine. With a distributed model, the more computation nodes available, the larger the problem that can be mapped.

4.2. Effects of resolution of input data

Vegetation cover datasets with different resolutions were used to test the performance of parallelization method. In addition to the spatial scale, the temporal scale can also vary, depending on the application of a model. For the dWaSSI-C model, the basic temporal scale has been fixed as a monthly time period. However, this is still a controversial and complex issue for ecological studies, because water and carbon processes at a specific scale are influenced by structure and function of other scales (Turnbull et al., 2008). In addition, these effects may even be non-linear, which changes the basic mathematical relationships for water and carbon processes during scaling (Yu et al., 2008). Thus, a mechanistic interpretation of the behaviour of a system can only be derived by an assessment of the extent to which ecosystem structure and function are connected through time and space (Turnbull et al., 2008).

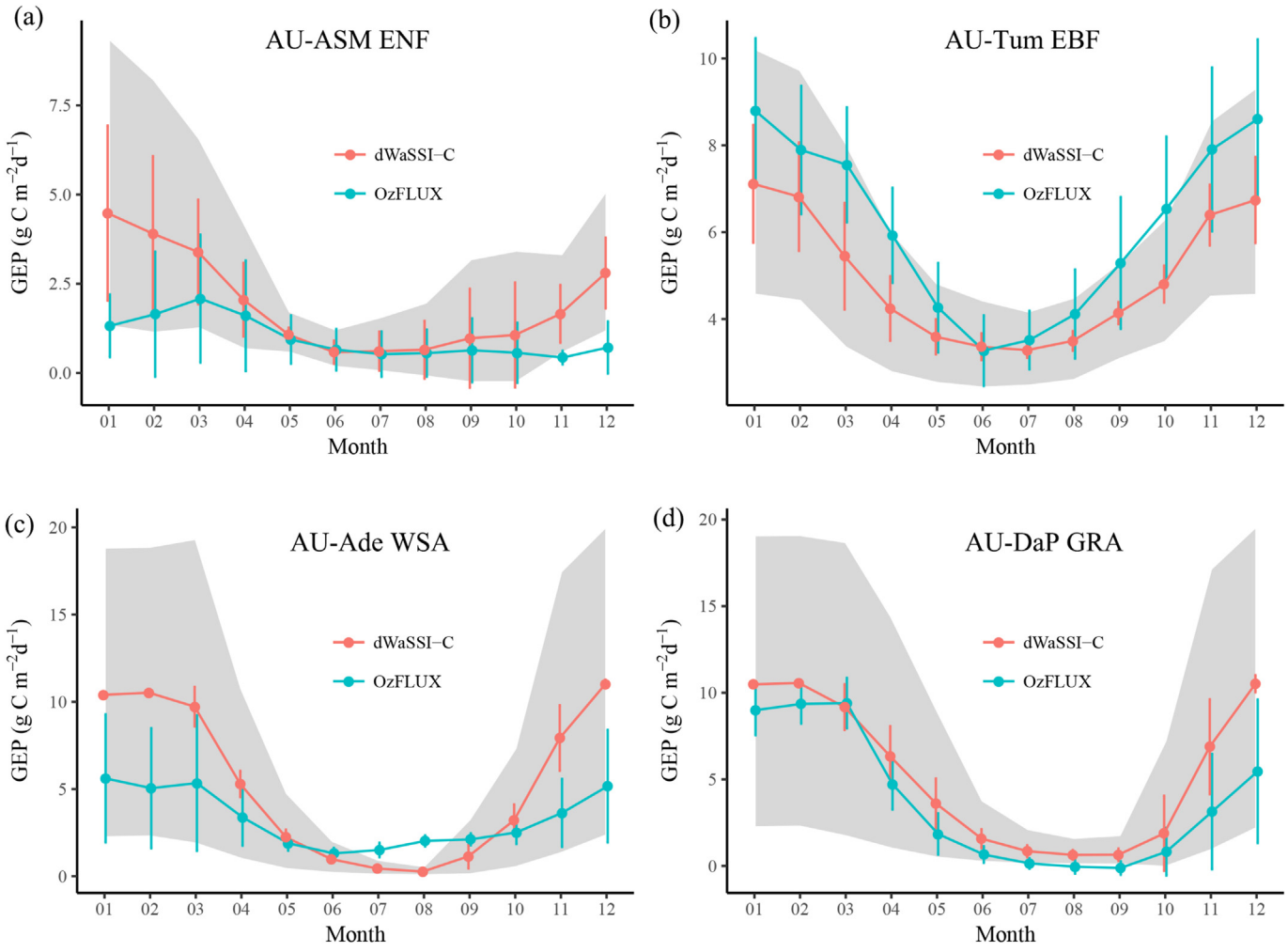

Fig. 11. Monthly mean of gross ecosystem productivity (GEP, g C m^{-2} d^{-1}) simulated by dWaSSI-C (red lines) and observed by OzFlux (blue lines) at (a) AU-ASM, (b) AU-Tum, (c) AU-Ade and (d) AU-DaP. The shaded region represents long-term (from 2000 to 2013) monthly mean GEP simulated by the dWaSSI-C model with WUE ± 1 * SD. Error bars on the lines represent one standard deviation for the monthly GEP during the period with observation data at each OzFlux site. ENF, EBF, WSA and GRA represent evergreen needleleaf forest, evergreen broadleaf forest, woody savannas and grassland, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

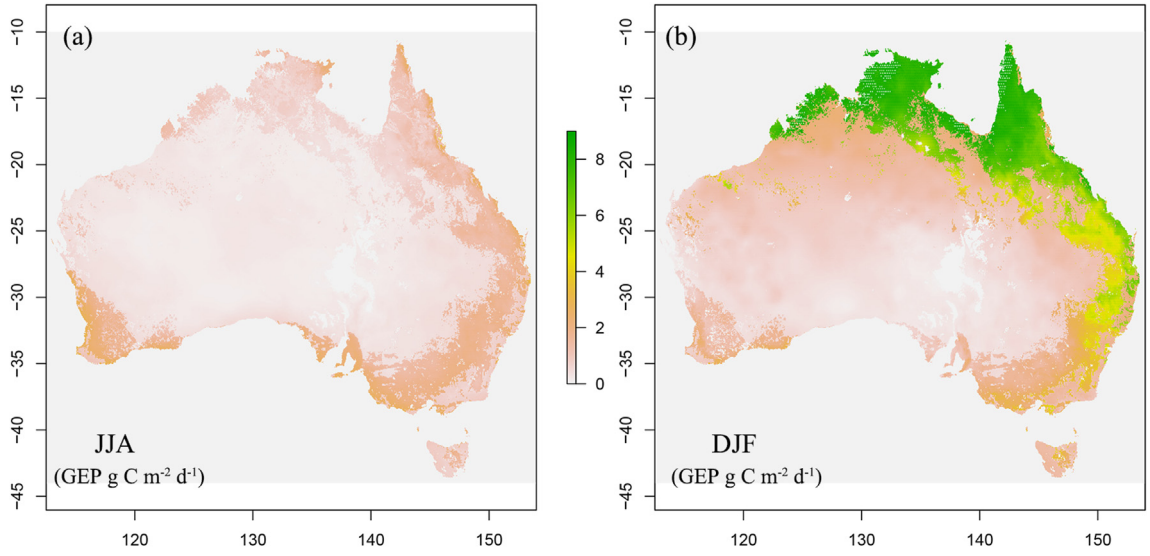

Fig. 12. Absolute differences in Gross Ecosystem Productivity (GEP, g C m^{-2} d^{-1}) in (a) winter (JJA) and (b) summer (DJF) between the control and control + 1 * SD of the WUE parameter.

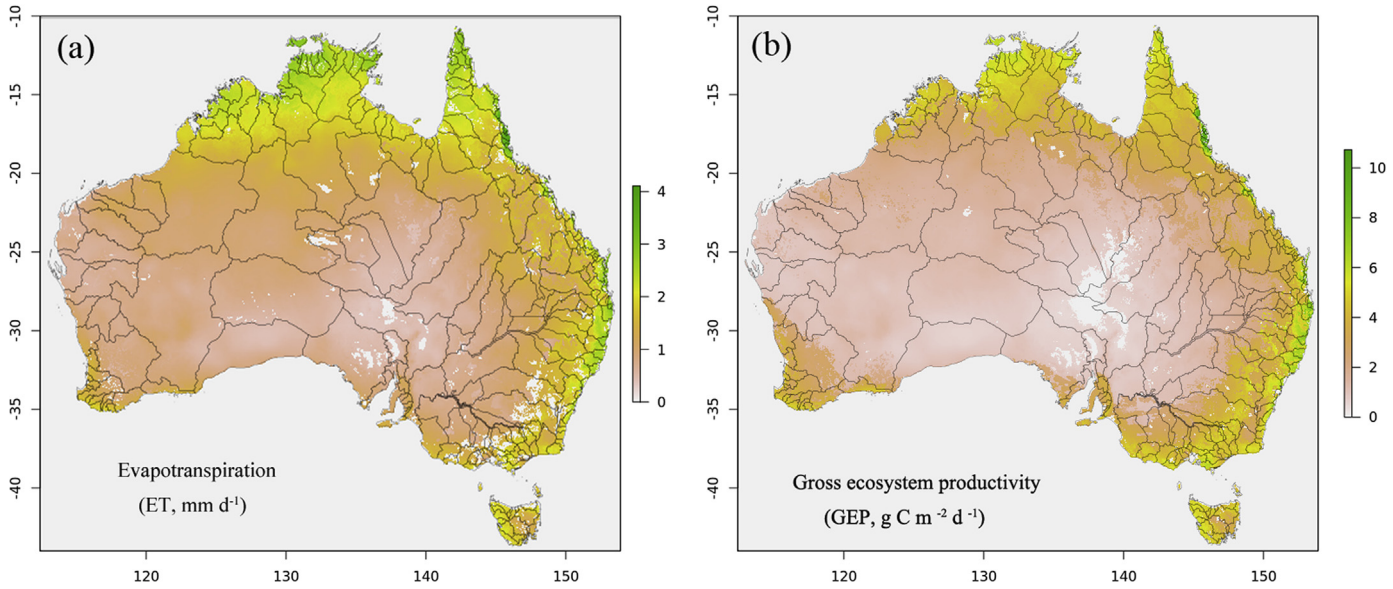

Fig. 13. Mean annual (a) evapotranspiration (ET, mm d^{-1}) and (b) gross ecosystem productivity (GEP, g C m^{-2} d^{-1}) estimates over the Australian continent from 5 * 5 km resolution dWaSSI-C simulations, averaged from 2000 to 2013.

In this study, we found that resolution of the input vegetation data-sets had little effect on water and carbon estimates by the dWaSSI-C model when the estimates were compared over the whole Australian continent. However, for each pixel, vegetation type is one of the most important factors influencing model estimates of water and carbon fluxes. Ecologically, vegetation spatial organization and constitution affect both water availability and carbon sequestration. Currently, the most commonly used remote sensing data used for terrestrial ecosystem detection are Landsat and MODIS, which have resolutions of 30 m and 500 m respectively. Even at the highest (30 m) resolution, each pixel may still represent combinations of several types of vegetation, hydrology, and soil characteristics, all of which can contribute to higher variability of fluxes within one pixel. Ma et al. (2015) found that a forest ecosystem carbon budget model for China at a 0.5 * 0.5° resolution, overestimated the forest gross carbon dioxide uptake by approximately 8.7% because the vegetation fraction per grid cell was not taken into consideration. Similarly, the dWaSSI-C model only considers one vegetation type per grid cell and hence, to accurately simulate water and carbon processes, the vegetation fraction per grid cell should be taken into consideration, and this is likely to add to the computational time.

Vegetation is a key factor which determines hydrological partitioning in a watershed and the consequences for watershed-scale hydrology. However, the role of vegetation in controlling the spatial and temporal dependence of water balance partitioning remains challenging to elucidate. Wood et al. (1988) suggests that there is a specific spatial resolution for each model at which point the model is insensitive to higher resolutions. Wolock (1995) found that the simulation accuracy of a topographical based hydrological model, TOPMODEL, increased with increasing resolution from 5*5 to 0.05*0.05 km in the Sleepers River Watershed, Vermont, USA. To resolve the difficulty of finding the most appropriate spatial resolution for hydrology models, Dehotin and Braud (2008) developed a nested discretization method, which allows a controlled and objective trade-off between available data, the resolution of the dominant water cycle components and the modeling objectives. Currently, dWaSSI-C has been only used for the watershed scale and its ideal simulation spatial resolution is still not clear. Therefore, using the parallelized version of dWaSSI-C, the ideal spatial resolution of dWaSSI-C and the effects of resolution of input data on the simulation can be more efficiently studied in the future.

4.3. Uncertainty of dWaSSI-C and future work

Although the dWaSSI-C model, when operated using default WUE parameters, can clearly capture the seasonal cycle of water and carbon fluxes in comparison with observations, a large area of Australia, such as the WSA and GRA areas, showed very high sensitivity of GEP to WUE, especially during the growing season (Figs. 10 and 11). These sensitivity results suggest that WUE is a very important and highly sensitive parameter for carbon flux estimations. Even when using the default WUE parameters, differences between the simulated and observed fluxes can be large, for example, during the first half of 2007 for WSA (Fig. 10). Given that there are observations available from around 30 sites from the OzFlux network (http://www.ozflux.org.au), and only two sites have been used for building the default model by Sun et al. (2011b), there is clearly scope for further work in better constraining the model using the latest available observational fluxes from the OzFlux network.

Another source of uncertainty in eco-hydrological models is the meteorological driving data (Slevin et al., 2017). The gridded climate data used in this study is originally interpolated from point observations of weather stations across the Australian continent. Therefore, the number of observations and the accuracy of the interpolation determine the quality of the climate data (Jones et al., 2009). A large portion of the inland arid Australian continent is poorly covered by meteorological stations, and hence, there is greater uncertainty in these regions (Jones et al., 2009). With the parallelized dWaSSI-C model, it is much easier and faster to conduct a model sensitivity analysis for the driving meteorological data.

Currently, all input and output data of dWaSSI-C model are stored in binary format, which has several disadvantages, such as the need for more processing to further analyze and visualize the data, and the need for additional documentation about the output binary format. Hierarchical Data Format (HDF) and Network

Common Data Format (NetCDF) provide a solution to these issues by enabling easier visualization and processing of model outputs, and have the added advantage that the meta-data are easily accessible from the output file, which avoids the need for additional documentation. Additionally, with NetCDF/HDF5 formats, the input and output datasets can be read and written in parallel, which would further reduce the compute time of the model. The performance of NetCDF/HDF5 formats will be analyzed as part of future research.

Acknowledgments

This work was supported by resources provided by the Pawsey Supercomputing Centre with funding from the Australian Government and the Government of Western Australia, a Murdoch University Doctoral Scholarship (Ning Liu) and the Chinese Academy of Forestry (Special Research Program for Public-welfare Forestry (201304201)). This work used eddy covariance data acquired and shared by the OzFlux-TERN network. Dr Jatin Kala is supported by an Australian Research Council Discovery Early Career Researcher Grant (DE170100102).

Abbreviations

The following abbreviations are used in this manuscript

WaSSI-C	Water Supply Stress Index and Carbon model
dWaSSI-C	distributed WaSSI-C
OpenMP	Open Multi-Processing
MPI	Message Passing Interface
I/O	input and output
SAC-SMA	Soil accounting content - Soil moisture availibity
WUE	Water use efficiency
ET	Evapotranspiration
PET	Potential evapotranspiration
GEP	Gross ecosystem productivity
ET_0	Reference evapotranspiration
ER	Ecosystem respiration
LAI	Leaf area index
MODIS	Moderate-resolution imaging spectroradiometer
MCD12Q1	The Land Cover Type Climate Modeling Grid (CMG) product of MODIS
IGBP	International Geosphere-Biosphere Programme
EBF	Evergreen broadleaf forest
ENF	Evergreen needleleaf forest
OSH	Open shrubland
WSA	Woody savannas
SA	Savannas
GRA	Grassland
CRO	Cropland

References

Anderson, R.M., Koren, V.I., Reed, S.M., 2006. Using SSURGO data to improve Sacramento Model *a priori* parameter estimates. J. Hydrology 320, 103–116. https://doi.org/10.1016/j.jhydrol.2005.07.020.

Baldocchi, D., Falge, E., Gu, L.H., Olson, R., Hollinger, D., Running, S., Anthoni, P., Bernhofer, C., Davis, K., Evans, R., Fuentes, J., Goldstein, A., Katul, G., Law, B., Lee, X.H., Malhi, Y., Meyers, T., Munger, W., Oechel, W., U, K.T.P., Pilegaard, K., Schmid, H.P., Valentini, R., Verma, S., Vesala, T., Wilson, K., Wofsy, S., 2001. FLUXNET: a new tool to study the temporal and spatial variability of ecosystem-scale carbon dioxide, water vapor, and energy flux densities. Bull. Am. Meteorological Soc. 82, 2415–2434. URL: <Go_to_ISI>://WOS:000171929700004. http://journals.ametsoc.org/doi/pdf/10.1175/1520-0477%282001%29082%3C2415%3AFANTTS%3E2.3.CO%3B2. doi:Doi 10.1175/1520-0477(2001)082<2415:Fantts>2.3.Co;2.

Beringer, J., 2013a. Adelaide River Ozflux Tower Site Ozflux: Australian and New Zealand Flux Research and Monitoring doi:102.100.100/14228.

Beringer, J., 2013b. Daly Pasture Ozflux Tower Site Ozflux: Australian and New Zealand Flux Research and Monitoring doi:102.100.100/14238.

Beringer, J., Hutley, L.B., McHugh, I., Arndt, S.K., Campbell, D., Cleugh, H.A., Cleverly, J., Resco de Dios, V., Eamus, D., Evans, B., Ewenz, C., Grace, P., Griebel, A., Haverd, V., Hinko-Najera, N., Huete, A., Isaac, P., Kanniah, K., Leuning, R., Liddell, M.J., Macfarlane, C., Meyer, W., Moore, C., Pendall, E., Phillips, A., Phillips, R.L., Prober, S.M., Restrepo-Coupe, N., Rutledge, S., Schroder, I., Silberstein, R., Southall, P., Yee, M.S., Tapper, N.J., van Gorsel, E., Vote, C., Walker, J., Wardlaw, T., 2016. An introduction to the Australian and New Zealand flux tower network – Ozflux. Biogeosciences 13, 5895–5916. https://doi.org/10.5194/bg-13-5895-2016. https://www.biogeosciences.net/13/5895/2016/.

Cao, M.K., Woodward, F.I., 1998. Dynamic responses of terrestrial ecosystem carbon cycling to global climate change. Nature 393, 249–252. https://doi.org/10.1038/30460. URL: <Go_toΩISI>://WOS:000073761000047.

Chen, L., Wang, L., Ma, Y., Liu, P., 2015. Overview of ecohydrological models and systems at the watershed scale. IEEE Syst. J. 9, 1091–1099. https://doi.org/10.1109/jsyst.2013.2296979.

Cleverly, J., 2011. Alice Springs Mulga Ozflux Tower Site Ozflux: Australian and New Zealand Flux Research and Monitoring doi:102.100.100/14217.

Dehotin, J., Braud, I., 2008. Which spatial discretization for distributed hydrological models? Proposition of a methodology and illustration for medium to large-scale catchments. Hydrology Earth Syst. Sci. Discuss. 12, 769–796.

Fatichi, S., Vivoni, E.R., Ogden, F.L., Ivanov, V.Y., Mirus, B., Gochis, D., Downer, C.W., Camporese, M., Davison, J.H., Ebel, B., Jones, N., Kim, J., Mascaro, G., Niswonger, R., Restrepo, P., Rigon, R., Shen, C., Sulis, M., Tarboton, D., 2016. An overview of current applications, challenges, and future trends in distributed process-based models in hydrology. J. Hydrology 537, 45–60. https://doi.org/10.1016/j.jhydrol.2016.03.026.

Friedl, M.A., Sulla-Menashe, D., Tan, B., Schneider, A., Ramankutty, N., Sibley, A., Huang, X.M., 2010. MODIS Collection 5 global land cover: algorithm refinements and characterization of new datasets. Remote Sens. Environ. 114, 168–182. https://doi.org/10.1016/J.Rse.2009.08.016. URL: <Go_toΩISI>://WOS:000271688700013.

van Gorsel, E., 2013. Tumbarumba Ozflux Tower Site Ozflux: Australian and New Zealand Flux Research and Monitoring doi:102.100.100/14241.

Hutchinson, M., 2014. Monthly Daily Maximum Temperature: ANUClimate 1.0, 0.01 Degree, Australian Coverage, 1970-2012. http://datamgt.nci.org.au:8080/geonetwork/srv/en/metadata.show?uuid=eMAST_ANUClimate_mon_tmax_v1m0_1970_2012. https://researchdata.ands.org.au/monthly-daily-maximum-1970-2012.

Jones, D.A., Wang, W., Fawcett, R., 2009. High-quality spatial climate data-sets for Australia. Aust. Meteorological Oceanogr. J. 58, 233–248. URL: <Go_to_ISI>://WOS:000274700500002.

Jonsson, P., Eklundh, L., 2004. TIMESAT-a program for analyzing time-series of satellite sensor data. Comput. Geosciences 30, 833–845. http://www.sciencedirect.com/science/article/pii/S0098300404000974. https://doi.org/10.1016/j.cageo.2004.05.006.

Kollet, S.J., Maxwell, R.M., Woodward, C.S., Smith, S., Vanderborght, J., Vereecken, H., Simmer, C., 2010. Proof of concept of regional scale hydrologic simulations at hydrologic resolution utilizing massively parallel computer resources. Water Resour. Res. 46 https://doi.org/10.1029/2009wr008730.

Le, P.V.V., Kumar, P., Valocchi, A.J., Dang, H.-V., 2015. GPU-based high-performance computing for integrated surfaceâĂŞsub-surface flow modeling. Environ. Model. Softw. 73, 1–13. https://doi.org/10.1016/j.envsoft.2015.07.015.

Li, T., Wang, G., Chen, J., Wang, H., 2011. Dynamic parallelization of hydrological model simulations. Environ. Model. Softw. 26, 1736–1746. https://doi.org/10.1016/j.envsoft.2011.07.015. http://www.sciencedirect.com/science/article/pii/S1364815211001769.

Liu, J., Chen, J.M., Cihlar, J., Park, W.M., 1997. A process-based boreal ecosystem productivity simulator using remote sensing inputs. Remote Sens. Environ. 62, 158–175. https://doi.org/10.1016/S0034-4257(97)00089-8. URL: <Go_toΩISI>://WOS: A1997YB44100004.

Liu, N., Sun, P.-S., Liu, S.-R., Sun, G., 2013a. Coupling simulation of water-carbon processes for catchment-calibration and validation of the WaSSI-C model. Chin. J. Plant Ecol. 37, 492–502. https://doi.org/10.3724/sp.j.1258.2013.00051.

Liu, N., Sun, P.-S., Liu, S.-R., Sun, G., 2013b. Determination of spatial scale of response unit for the WASSI-C eco-hydrological model-a case study on the upper Zagunao River watershed of China. Chin. J. Plant Ecol. 37, 132–141. https://doi.org/10.3724/sp.j.1258.2013.00014.

Ma, J., Yan, X., Dong, W., Chou, J., 2015. Gross primary production of global forest ecosystems has been overestimated. Sci. Rep. 5, 10820. https://doi.org/10.1038/srep10820. https://www.ncbi.nlm.nih.gov/pubmed/26027557.

McKenzie, N., Jacquier, D., Ashton, L., Cresswell, H., 2000. Estimation of soil properties using the Atlas of Australian Soils. CSIRO Land Water Tech. Rep. 11, 1–12.

Raczka, B.M., Davis, K.J., Huntzinger, D., Neilson, R.P., Poulter, B., Richardson, A.D., Xiao, J., Baker, I., Ciais, P., Keenan, T.F., Law, B., Post, W.M., Ricciuto, D., Schaefer, K., Tian, H., Tomelleri, E., Verbeeck, H., Viovy, N., 2013. Evaluation of continental carbon cycle simulations with North American flux tower observations. Ecol. Monogr. 83, 531–556. https://doi.org/10.1890/12-0893.1.

Sjostrom, M., Zhao, M., Archibald, S., Arneth, A., Cappelaere, B., Falk, U., de Grandcourt, A., Hanan, N., Kergoat, L., Kutsch, W., Merbold, L., Mougin, E., Nickless, A., Nouvellon, Y., Scholes, R.J., Veenendaal, E.M., Ardo, J., 2013. Evaluation of MODIS gross primary productivity for Africa using eddy covariance data. Remote Sens. Environ. 131, 275–286. https://doi.org/10.1016/

J.Rse.2012.12.023. URL: <Go_toΩISI>://WOS:000315546900021.
Slevin, D., Tett, S.F.B., Exbrayat, J.-F., Bloom, A.A., Williams, M., 2017. Global evaluation of gross primary productivity in the JULES land surface model v3.4.1. Geosci. Model Dev. 10, 2651–2670. https://doi.org/10.5194/gmd-10-2651-2017. https://www.geosci-model-dev.net/10/2651/2017/.
Stoy, P.C., Katul, G.G., Siqueira, M.B.S., Juang, J.Y., Novick, K.A., Uebelherr, J.M., Oren, R., 2006. An evaluation of models for partitioning eddy covariance-measured net ecosystem exchange into photosynthesis and respiration. Agric. For. Meteorology 141, 2–18. http://www.scopus.com/inward/record.url?eid=2-s2.0-33845233178&partnerID=40&md5=5242ce60d2ec29fec58af2b9f4e6d038.
Sun, G., Alstad, K., Chen, J.Q., Chen, S.P., Ford, C.R., Lin, G.H., Liu, C.F., Lu, N., McNulty, S.G., Miao, H.X., Noormets, A., Vose, J.M., Wilske, B., Zeppel, M., Zhang, Y., Zhang, Z.Q., 2011a. A general predictive model for estimating monthly ecosystem evapotranspiration. Ecohydrology 4, 245–255. https://doi.org/10.1002/Eco.194. URL: <Go_toΩISI>://WOS:000289263900010.
Sun, G., Caldwell, P., Noormets, A., McNulty, S.G., Cohen, E., Moore Myers, J., Domec, J.-C., Treasure, E., Mu, Q., Xiao, J., John, R., Chen, J., 2011b. Upscaling key ecosystem functions across the conterminous United States by a water-centric ecosystem model. J. Geophys. Res. 116, 1–16. https://doi.org/10.1029/2010JG001573.
Sun, S., Sun, G., Caldwell, P., McNulty, S.G., Cohen, E., Xiao, J., Zhang, Y., 2015. Drought impacts on ecosystem functions of the U.S. National Forests and Grasslands: Part I evaluation of a water and carbon balance model. For. Ecol. Manag. 353, 260–268. https://doi.org/10.1016/j.foreco.2015.03.054.
Taylor, G.T., Muller-Karger, F.E., Thunell, R.C., Scranton, M.I., Astor, Y., Varela, R., Ghinaglia, L.T., Lorenzoni, L., Fanning, K.A., Hameed, S., Doherty, O., 2012. Ecosystem responses in the southern Caribbean Sea to global climate change. Proc. Natl. Acad. Sci. 109, 19315–19320. https://doi.org/10.1073/pnas.1207514109. http://www.pnas.org/content/109/47/19315.abstract. http://www.pnas.org/content/109/47/19315.full.pdf.
Trudinger, C.M., Haverd, V., Briggs, P.R., Canadell, J.G., 2016. Interannual variability in Australia's terrestrial carbon cycle constrained by multiple observation types. Biogeosciences 13, 6363–6383. https://doi.org/10.5194/bg-13-6363-2016. URL: <Go_toΩISI>://WOS:000388715700001.
Turnbull, L., Wainwright, J., Brazier, R.E., 2008. A conceptual framework for understanding semi-arid land degradation: ecohydrological interactions across multiple-space and time scales. Ecohydrology 1, 23–34. https://doi.org/10.1002/eco.4.
Wolock, D.M., 1995. Effects of subbasin size on topographic characteristics and simulated flow paths in Sleepers River watershed, Vermont. Water Resour. Res. 31, 1989–1997. https://doi.org/10.1029/95wr01183. URL: <Go_toΩISI>://WOS:A1995RM64400014.
Wood, E.F., Roundy, J.K., Troy, T.J., van Beek, L.P.H., Bierkens, M.F.P., Blyth, E., de Roo, A., DÃűll, P., Ek, M., Famiglietti, J., Gochis, D., van de Giesen, N., Houser, P., JaffÃl, P.R., Kollet, S., Lehner, B., Lettenmaier, D.P., Peters-Lidard, C., Sivapalan, M., Sheffield, J., Wade, A., Whitehead, P., 2011. Hyperresolution global land surface modeling: meeting a grand challenge for monitoring Earth's terrestrial water. Water Resour. Res. 47 https://doi.org/10.1029/2010wr010090.
Wood, E.F., Sivapalan, M., Beven, K., Band, L., 1988. Effects of spatial variability and scale with implications to hydrologic modeling. J. Hydrology 102, 29–47. https://doi.org/10.1016/0022-1694(88)90090-X. URL: <Go_toΩISI>://WOS:A1988Q618100003.
Xiao, J., Chen, J., Davis, K.J., Reichstein, M., 2012. Advances in upscaling of eddy covariance measurements of carbon and water fluxes. J. Geophys. Res. Biogeosciences 117. https://doi.org/10.1029/2011jg001889.
Xiao, J., Sun, G., Chen, J., Chen, H., Chen, S., Dong, G., Gao, S., Guo, H., Guo, J., Han, S., Kato, T., Li, Y., Lin, G., Lu, W., Ma, M., McNulty, S., Shao, C., Wang, X., Xie, X., Zhang, X., Zhang, Z., Zhao, B., Zhou, G., Zhou, J., 2013. Carbon fluxes, evapotranspiration, and water use efficiency of terrestrial ecosystems in China. Agric. For. Meteorology 76–90. https://doi.org/10.1016/j.agrformet.2013.08.007, 182-183.
Yu, G., Song, X., Wang, Q., Liu, Y., Guan, D., Yan, J., Sun, X., Zhang, L., Wen, X., 2008. Water-use efficiency of forest ecosystems in eastern China and its relations to climatic variables. New Phytol. 177, 927–937. https://doi.org/10.1111/j.1469-8137.2007.02316.x. http://www.ncbi.nlm.nih.gov/pubmed/18069958. http://onlinelibrary.wiley.com/store/10.1111/j.1469-8137.2007.02316.x/asset/j.1469-8137.2007.02316.x.pdf?v=1&t=i0631bg3&s=5744da26200b294f75d6be24d7629d03bcbcc908.
Yuan, H., Dai, Y., Xiao, Z., Ji, D., Shangguan, W., 2011. Reprocessing the MODIS Leaf Area Index products for land surface and climate modelling. Remote Sens. Environ. 115, 1171–1187.
Zhang, A., Li, T., Si, Y., Liu, R., Shi, H., Li, X., Li, J., Wu, X., 2016. Double-layer parallelization for hydrological model calibration on HPC systems. J. Hydrology 535, 737–747. https://doi.org/10.1016/j.jhydrol.2016.01.024. http://www.sciencedirect.com/science/article/pii/S002216941600041X.
Zhao, M., Heinsch, F.A., Nemani, R.R., Running, S.W., 2005. Improvements of the MODIS terrestrial gross and net primary production global data set. Remote Sens. Environ. 95, 164–176. https://doi.org/10.1016/j.rse.2004.12.011.
Zhou, X., Bi, S., Yang, Y., Tian, F., Ren, D., 2014. Comparison of ET estimations by the three-temperature model, SEBAL model and eddy covariance observations. J. Hydrology 519, 769–776. https://doi.org/10.1016/j.jhydrol.2014.08.004. http://www.sciencedirect.com/science/article/pii/S0022169414006027. http://ac.els-cdn.com/S0022169414006027/1-s2.0-S0022169414006027-main.pdf?_tid=c5941cb0-2366-11e4-8444-00000aab0f01&acdnat=1407988683_a0b3632098db1e9ff671a1498f597abc.