










APPLICATION

hydrographr: An R package for scalable hydrographic data processing

Marlene Schürz^{1,2} | Afroditi Grigoropoulou^{1,2}  | Jaime García Márquez¹  |
 Yusdiel Torres-Cambas¹  | Thomas Tomiczek¹ | Mathieu Floury¹  |
 Vanessa Bremerich¹  | Christoph Schürz³  | Giuseppe Amatulli^{1,4,5}  |
 Hans-Peter Grossart^{6,7}  | Sami Domisch¹ 

¹Department of Community and Ecosystem Ecology, Leibniz Institute of Freshwater Ecology and Inland Fisheries (IGB), Berlin, Germany; ²Department of Biology, Chemistry, Pharmacy, Institute of Biology, Freie Universität Berlin, Berlin, Germany; ³Department of Computational Landscape Ecology, UFZ-Helmholtz-Centre for Environmental Research, Leipzig, Germany; ⁴School of the Environment, Yale University, Connecticut, New Haven, USA; ⁵Spatial Ecology, Penn, UK; ⁶Plankton and Microbial Ecology, Leibniz Institute for Freshwater Ecology and Inland Fisheries (IGB), Berlin, Germany and ⁷University of Potsdam Institute of Biochemistry and Biology, Potsdam, Germany

Correspondence

Marlene Schürz
 Email: marlene.schuerz@igb-berlin.de

Afroditi Grigoropoulou
 Email: afroditi.grigoropoulou@igb-berlin.de

Sami Domisch
 Email: sami.domisch@igb-berlin.de

Funding information

Leibniz Competition, Grant/Award Number: J45/2018; Alexander von Humboldt Foundation, Grant/Award Number: Ref 3.2-CUB-1212347-GF-P; German Federal Ministry of Education and Research, Grant/Award Number: 033W034A; NFDI4Biodiversity, German Research Foundation, Grant/Award Number: 442032008; NFDI4Earth, German Research Foundation, Grant/Award Number: 460036893

Handling Editor: David Soto

Abstract

1. Freshwater ecosystems are considered biodiversity hotspots, but assessing the spatial distribution of species remains challenging. One major obstacle lies in the complex geospatial processing of large amounts of data, such as stream network, sub-catchment and basin data, that are necessary for addressing the longitudinal connectivity among water bodies. Workflows thus need to be scalable, especially when working across large spatial extents and at high spatial resolution. This in turn requires advanced command-line GIS skills and programming language integration, which often poses a challenge for freshwater researchers.
2. To address this challenge, we developed the package `hydrographr` that provides scalable hydrographic data processing in R. The package contains functions for downloading data of the high-resolution Hydrography90m dataset, processing, reading and extracting information, as well as assessing network distances and connectivity. While the functions are, by default, tailored toward the Hydrography90m data, they can also be generalised toward other data and purposes, such as efficient cropping and merging of raster and vector data, point-raster extraction, raster reclassification and data aggregation. The package depends on the open-source software GDAL/OGR, GRASS-GIS and the AWK programming language in the Linux environment, allowing a seamless language integration. Since the data is processed outside R, `hydrographr` allows creating scalable geo-processing workflows.

Marlene Schürz and Afroditi Grigoropoulou contributed equally.

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2023 The Authors. *Methods in Ecology and Evolution* published by John Wiley & Sons Ltd on behalf of British Ecological Society.

3. We illustrate the `hydrographr` functions using two workflows that focus on (i) a freshwater species distribution modelling approach, and (ii) assessing stream connectivity given the fragmentation by dams. We also provide a detailed guide for the initial installation of the required software. Windows users need to first enable the Windows Subsystem for Linux (WSL) feature, and can then follow the same software installation as Linux users. `hydrographr` is maintained on GitHub at <https://github.com/flowabio/hydrographr>.
4. `hydrographr` provides a set of key functions for processing freshwater geospatial data. We expect that the package will support the freshwater-related research communities given the easy-to-use wrapper functions that allow capitalizing on powerful open-source command-line software, which may otherwise require a steep learning curve. Users can thus perform large-scale freshwater-specific longitudinal connectivity and network analyses across large geographic extents while staying within the R environment.

KEYWORDS

connectivity, hydrographic data processing, Hydrography90m, river, R-package, scalability, spatial freshwater biodiversity, stream network

1 | INTRODUCTION

Freshwater ecosystems harbour a significant proportion of biodiversity given the fraction of the Earth's surface they cover, while the high prevalence of endemic species renders them disproportionately vulnerable to biodiversity loss (Reid et al., 2019). The substantial knowledge gaps in spatial freshwater biodiversity research, currently hindering consensus on global biodiversity trends (Jähnig et al., 2021; van Klink et al., 2020), stem primarily from two factors: the lack of publicly available species occurrence information globally (Meyer et al., 2015), and the complex geospatial data processing that differs significantly from the terrestrial and marine realms, given the stream network characteristics (Altermatt, 2013).

To address this challenge, it is important to provide spatial analysis tools that improve the estimation of species distributions and patterns of freshwater organisms. Freshwater ecosystems are characterised by the unique longitudinal connectivity within a dendritic network structure. Given the high degree of spatial autocorrelation across the network, spatial biodiversity analyses need to incorporate connectivity between stream locations, which often requires non-trivial geospatial data processing workflows (Ver Hoef et al., 2014). In freshwater geospatial analyses, e.g., related to biodiversity, water quality/quantity or river fragmentation, the processing of the spatial information prior to the analyses typically involves (i) delineating a stream network and corresponding catchments, (ii) moving (“snapping”) species point coordinates to the stream network given possible spatial mismatches between a modelled stream network and the GPS-derived sampling locations and (iii) extracting environmental, or species co-occurrence information across the network or corresponding sub-catchments, which form the spatial units of the analyses (Domisch et al., 2015). Further steps could include measuring Euclidean (“as-the-crow-flies”) and network (“as-the-fish-swims”) distances among species occurrences, while accounting for dams and weirs, to analyse population connectivity within the river network.

Such analyses are computationally demanding and time-consuming when carried out for an extensive dataset of species records or at large spatial extents. For instance, catchment-scale analyses (i.e. network delineation, or aggregating catchment environmental features) in downstream areas may require information from the upstream contributing area. Consequently, the resulting size of the study area that needs to be taken into account can be considerably larger than the focal sampling area. In addition, assessing freshwater ecosystems requires a high spatial resolution to also include small water bodies, as opposed to lumping the heterogeneous freshwater environment into larger catchment basins as spatial units. In this regard, the recently published Hydrography90m dataset (Amatulli et al., 2022) addresses such small streams with unprecedented spatial detail, providing a globally seamless and standardised hydrographic dataset (S1.1. Supplementary information; The Hydrography90m dataset). The amount of spatial data, however, translating into 1.6 million drainage basins and 726 million stream segments, which serve as the spatial units, requires efficient processing workflows to support the research communities in performing high-resolution spatial analyses in freshwater ecosystems.

1.1 | Tools for spatial freshwater data processing

1.1 | Tools for spatial freshwater data processing

Most of the freshwater geospatial data processing is done using graphical user interface (GUI) software, such as QGIS (QGIS Development Team, 2021) and ArcGIS (Redlands, 2011). They provide essential tools, but they cannot be easily scaled toward larger amounts of data or remain proprietary and hence out of reach. Opposed to GUI-based

software, command-line based open-source software and libraries are optimised toward reproducibility and interoperability for creating flexible workflows. In this regard, the popularity of the R software is increasing in an inexorable way, to the point that it has become the lingua franca in many research disciplines (Lai et al., 2019). In particular, R is widely used in spatial data processing given R packages such as terra (Hijmans, 2023) or sf (Pebesma, 2018).

While R is undoubtedly versatile, it is not optimised toward scalability, as objects are required to be loaded into the R workspace (RAM of the computer), which can potentially pose constraints for ordinary computers. In contrast, other open-source command-line software, such as the Geospatial Data Abstraction Library (GDAL Development Team, 2020) and Geographic Resources Analysis Support System (GRASS GIS; GRASS Development Team, 2022), are optimised toward efficiency and scalability, given that they enable multi-thread processing of very large datasets due to an optimised memory management (Amatulli et al., 2014). Nevertheless, they come at the expense of a steep learning curve in terms of syntax and workflows. Moreover, the seamless interoperability of these tools, for example, supported by the native Bash (GNU, 2007) language and integrated command-line utilities such as AWK (Aho et al., 1979) and sed (Pizzini et al., 2018) for manipulating large tables, is mainly provided within the Linux operating system (OS), adding another hurdle for potential users.

We aimed to address this problem by developing the `hydrographr` R-package, which combines the two key requirements from a user perspective: (i) users can capitalise on the entire suite of the seamlessly integrated and RAM-efficient open-source software that manipulates the data directly on disk, while (ii) staying in the widely used working R environment (R Core Team, 2022) without the need to learn the syntax of various software.

2 | THE `hydrographr` R-PACKAGE

The `hydrographr` R-package provides R functions that serve as wrappers for GDAL/OGR and GRASS GIS functions to efficiently work with the Hydrography90m spatial data. The package is written in the R and Bash languages. On a Windows OS, the Bash language is interpreted using Windows commands. It currently includes 21 functions and uses a variety of spatial and non-spatial data processing tools provided in the [Supporting Information S1.2](#). Supplementary information; The `hydrographr` R-package dependencies. The advantage of `hydrographr` is its memory-efficiency by avoiding loading the data into the R workspace during processing, and the creation of temporary data on disc. The functions ([Table 1](#)) allow to

- download the Hydrography90m data,
- process and extract information from spatial layers without explicitly loading them into R,
- read data into R while applying spatial filtering,
- calculate distances between points (e.g. species occurrence and dams) and obtain information related to stream and basin fragmentation, and

- capitalise on the speed and versatility of network graphs to assess network connectivity.

While the default settings of the functions are tailored toward Hydrography90m, they are also generalizable to process other hydrographic data and non-hydrographic data ([S1.3](#). Supplementary information; Usage with other spatial datasets, [S4](#). Other_stream_network) directly on disc. This includes cropping and merging of raster and vector data, point-raster extraction, raster reclassification and data aggregation (see [Table 1](#)).

2.1 | Downloading the Hydrography90m data

The Hydrography90m data is organised in a regular and irregular tiling system (RTS and ITS, respectively). In the RTS, the data is divided into $20^{\circ} \times 20^{\circ}$ tiles, optimised toward downloading but with an interrupted across-basin connectivity. In the ITS, the data is divided in *regional units* which encompass entire drainage basins (Amatulli et al., 2022) without any interruptions in connectivity, but resulting in very large files (e.g. the Amazon basin). The functions `get_tile_id` and `get_regional_unit_id` situate a data frame with WGS84 coordinates in the corresponding regular tile or regional unit and return the unit IDs. These IDs can be then passed to the function `download_tiles` which downloads the data. We also provide the function `download_test_data` that downloads species occurrence data and spatial layers from a small study area, useful for running the examples of the package manual.

2.2 | Spatial data processing

`hydrographr` offers the possibility to process spatial layers without loading them into R. The functions `merge_tiles` and `crop_to_extent` can be used to merge raster (.tif) and vector (.gpkg) and crop raster files to a given extent respectively, while the function `set_no_data` modifies the NoData value of raster layers. The two snapping functions `snap_to_network` and `snap_to_subc_segment` snap point locations (e.g. species occurrences) to the digital stream network, meaning that the coordinates of the point locations get slightly changed so that the point overlaps with the stream segment. While the function `snap_to_network` snaps the point within a given distance radius and, if given, also only to stream segments with a minimum flow accumulation, the `snap_to_subc_segment` function snaps the point to the stream segment of the sub-catchment where it is located. Finally, `reclass_raster` reclassifies a raster following rules provided in a data frame.

2.3 | File reading and data extraction

This group of functions extracts information from raster or vector files. For instance, the function `extract_ids` receives WGS84

TABLE 1 The functions of `hydrographr` for downloading, processing data, reading and extracting data, distance related and graph-based connectivity analyses, grouped by category and listing the function name along with an icon, and the description. Terminology of the Hydrography90m (Amatulli et al., 2022): *Regional unit*: a rectangular area that contains only entire drainage basins, masking any truncated ones, useful for selecting a drainage basin within study areas. *Drainage basin*: any area of land where precipitation collects and drains into a common outlet. The outlet can be at the coast or an inland depression. *Stream segment*: the stream channel between two segment nodes (or from initialisation to the first confluence) of the network. *Sub-catchment*: land area between two segment nodes that contributes to the local flow accumulation of a given stream segment. Sub-catchments and stream segments share a unique ID.


















Category	Function name	Description
Downloading	 <code>get_tile_id</code>	Identifies the ID of the regular tile(s) of the Hydrography90m, where the input points are located. The output IDs are required to download the data using the function <code>download_tiles</code>
	 <code>get_regional_unit_id</code>	Identifies the ID of the regional unit(s) of the Hydrography90m, where the input points are located. The output IDs are required to download the data using the function <code>download_tiles</code>
	 <code>download_tiles</code>	Downloads data of the Hydrography90m dataset
	 <code>download_test_data</code>	Downloads the test data of the Hydrography90m dataset, required to run the examples of the manual
Processing	 <code>merge_tiles</code>	Merges raster (.tif) or vector (.gpkg) files using the GDAL functions <code>gdalbuildvrt</code> and <code>gdal_translate</code> for raster files and <code>ogrmerge.py</code> and <code>ogr2ogr</code> for vector files.
	 <code>crop_to_extent</code>	Crops a raster (.tif) file to a polygon border line or to the extent of a bounding box using the GDAL function <code>gdalwarp</code>
	 <code>snap_to_network</code>	Snaps points to the next stream segment within a defined radius, or a defined radius and a minimum flow accumulation using the GRASS GIS function <code>r.stream.snap</code>
	 <code>snap_to_sub_catchment</code>	Snaps points to the stream segment of the sub-catchment where the points are located in using the GRASS GIS functions <code>v.net</code> and <code>v.distance</code>
	 <code>set_no_data</code>	Sets a NoData value to input files using the GDAL function <code>gdal_edit.py</code>
	 <code>reclass_raster</code>	Reclassifies an integer raster (.tif) layer using the GRASS GIS function <code>r.reclass</code>
Reading & data extraction	 <code>extract_ids</code>	Extracts the ID value of the drainage basin and/or sub-catchment raster layer at given point locations using the GDAL function <code>gdalallocationinfo</code>
	 <code>report_no_data</code>	Reports the NoData value of input files using the GDAL function <code>gdalinfo</code>
	 <code>extract_zonal_stat</code>	Calculates zonal statistics based on one or more environmental variable raster .tif layers across a set (or all) of sub-catchments in a spatial extent using the GRASS GIS function <code>r.univar</code>
	 <code>read_geopackage</code>	Loads a .gpkg file, or only a part of it, as a <code>data.table</code> , <code>graph</code> , <code>sf</code> , or <code>SpatVector</code> object
	 <code>get_upstream_catchment</code>	Calculates the upstream basin, taking each point as the outlet using the GRASS GIS function <code>g.region</code>

TABLE 1 (Continued)

Distance-related		<code>get_distance</code>	Calculates the euclidean or within-network distance between points using the GRASS GIS function <code>v.distance</code> or <code>v.net.all.pairs</code> . To parallelize calculations per basin, see also <code>get_distance_parallel</code>
Graph-based connectivity analyses		<code>get_segment_neighbours</code>	Reports the up- and/or downstream stream segments that are connected to the input segments within a neighbour order. Provides the option to summarise attributes across these segments
		<code>get_catchment_graph</code>	Extracts the upstream, downstream, or entire catchment of the input stream segments from a network graph
		<code>get_distance_graph</code>	Calculates the network distance between all input sub-catchment IDs from node to node (outlet of the stream segment)
		<code>get_pfafstetter_basins</code>	Delineates Pfafstetter sub-basins for the input stream network

coordinates and extracts the IDs of the sub-catchments or drainage basins where the points are located, given the corresponding raster layers. In fact, the function can be used to extract values at point locations given any input raster layer. The function `extract_zonal_stat` calculates statistics on cell values of a raster within the zones defined by, for example a sub-catchment or other zonal raster layer. The function `read_geopackage` imports a vector file as a table, directed graph or spatial object, allowing for spatial filtering by IDs. For example, in the case of the stream order vector files of the Hydrography90m, only some streams can be imported based on specific sub-catchment IDs. In case the analyses need to be limited to the upstream area of a point location, `get_upstream_catchment` calculates the upstream basin of a given point, considering it as the outlet. Finally, the function `report_no_data` reports the NoData value of raster layers, which is crucial for calculating summary statistics.

2.4 | Distance calculations

The function `get_distance` calculates Euclidean distances or distances along the stream network between all pairs of point locations (e.g. occurrence points or dam locations). In case points are spread across multiple basins, `get_distance_parallel` parallelises the distance computations across drainage basins.

2.5 | Graph-based connectivity analyses

The function `get_segment_neighbours` queries the stream segments, or corresponding sub-catchments, that are first- or higher-order neighbours of a given segment (i.e. directly vs. indirectly adjacent, respectively), according to the user-specified degree. The function returns a list of data tables, containing the neighbouring segments, for each input segment ID. These can be

useful for spatially explicit models or, for example in conservation planning analyses (Domisch et al., 2019) which integrate the network connectivity. `get_catchment_graph` delineates the up and/or downstream catchment for the input segment IDs. The function returns either a directed graph or a data table with the IDs of the sub-catchments. The output is useful to, for example, restrict the focal area only to those parts of the stream network that are connected to a set of points. `get_distance_graph` performs the along-the-network distance analysis of points on a network graph and returns either (i) the pair-wise network distance (in meters) between all input points, or (ii) the number of network segments (i.e. sub-catchments) that are along the paths. The distances are calculated between the nodes of those segments where the points are located (opposed to the exact location on the given segment). `get_pfafstetter_basins` follows Verdin and Verdin (1999) and hierarchically subdivides a given drainage basin in up to nine smaller sub-basins. For this purpose, the function finds the most-contributing stream within the basin based on flow accumulation, and generates sub-basins along this main stream course. The sub-basins may serve as new spatial units to tackle, for example, scale-dependent effects given the Modifiable Area Unit Problem (MAUP; Fotheringham & Wong, 1991) when aggregating data across spatial units along the stream network.

2.6 | Software requirements

`hydrographr` relies on the open-source GRASS GIS software, GDAL/OGR library and AWK programming language within the Linux OS. While Linux users can install these tools directly in their shell, Windows users need to first activate the Windows Subsystem for Linux (WSL) that requires Windows 10 v.2004 or higher. The WSL allows running a Linux OS on Windows without a virtual machine or a dual boot system. We opted for using Linux as the main system in `hydrographr` given the seamless language integration

among GRASS GIS, GDAL/OGR, AWK and R, which ultimately increases the scalability of workflows. Moreover, from the variety of software, we could choose exactly those tools that perform best at a given task. `hydrographr` requires GRASS GIS 8.2 or higher, GDAL/OGR 3.4.3 or higher, GNU parallel (Tange et al., 2011) 20210822 and bc 1.07.1 or higher and for Windows users dos2unix. For further information about the software see [Supporting Information S1.2](#). A step-by-step installation manual can be found at https://glowabio.github.io/hydrographr/articles/linux_system_setup.html for Linux users and https://glowabio.github.io/hydrographr/articles/windows_system_setup.html for Windows users.

3 | CASE STUDIES

We present two case studies to showcase the functionality of the `hydrographr` package (i) using a species distribution modelling (SDM) workflow and (ii) in a connectivity analysis framework to estimate the free-flowing river length between species observations and dams. Throughout, we refer to [Table 1](#) for the description of the functions, [Figure 1](#) for a visualization of the example workflows and https://glowabio.github.io/hydrographr/articles/case_study_cuba.html and https://glowabio.github.io/hydrographr/articles/case_study_brazil.html for the actual code to reproduce the workflows (see also [Supporting Information S2](#). `Case_study_1` and `Case_study_2`).

3.1 | Species distribution modelling workflow

In the first case study ([Figure 1a](#), [Supporting Information S2](#). `Case_study_1`), we use a simple SDM to predict the future suitable habitats of the dragonfly species *Hypolestes trinitatis* in Cuba under a climate change scenario.

We obtained georeferenced occurrence points of the species from the dataset “Ephemeroptera, Trichoptera and Odonata of Cuba” (Torres-Cambas & Salazar Salina, 2022). The hydromorphological variable layers drawn from the Hydrography90m were the maximum linear curvature along the watercourse of each sub-catchment to approximate hydraulic complexity, the stream power index (spi), which represents the erosive power associated with flow (Moore et al., 1991), in raster format, and the length of the stream within each sub-catchment, in vector format. We identified the IDs of the two regular 20°×20° tiles of the Hydrography90m where the occurrence points were located using the function `get_tile_id` and downloaded the layers using `download_tiles`. Additionally, we downloaded the global layers of the CHELSA bioclimatic variables of mean annual air temperature, annual precipitation amount and precipitation seasonality. Specifically, we used the 30-year average of 1981–2010 and 2041–2070, as predicted by the IPSL-CM6A-LR climate model under the scenario SSP370, to describe the present and future climate, respectively (Karger & Zimmermann, 2019). We cropped the raster layers using the

function `crop_to_extent` and merged the Hydrography90m tiles using `merge_tiles`.

The spatial units of the analysis were the sub-catchments, and therefore all the variables were aggregated per sub-catchment. We obtained the zonal statistics (mean and standard deviation) of the maximum channel curvature, spi and bioclimatic variables within each sub-catchment using the function `extract_zonal_stat` and extracted the absolute value of stream length from the stream order .gpkg file, using `read_geopackage`. We first obtained these values for all the sub-catchments of the study area, and then extracted the sub-catchments where the species occurred based on their IDs, which we acquired using `extract_ids`, as well as 10,000 randomly sampled sub-catchments that served as the pseudoabsences in the model. We fit a down-sampled (Valavi et al., 2021) classification random forest using the ‘ranger’ package and obtained the model's predictions through the `predict` function (Wright & Ziegler, 2017). Finally, we provided a table containing the predicted habitat suitability for every sub-catchment ID as input to the `reclass_raster` function, to reclassify the sub-catchment raster of the modelling domain, and map the future habitat suitability.

3.2 | Connectivity analysis workflow

In the second case study ([Figure 1b](#), [Supporting Information S3](#). `Case_study_2`), we showcase the `hydrographr` functions by estimating the possible future change in the free-flowing river length between species occurrences and dam locations. We focused on the migrating catfish *Conorhynchos conirostis*, endemic to the São Francisco drainage basin in Brazil. The species is considered endangered (IUCN, 2022), being mostly affected by dams.

In short, the task was to identify (i) which dams are located up- and downstream of each fish occurrence and (ii) to evaluate the shortest distance from each occurrence to the closest existing and future up- and downstream dam. While some processing steps could be potentially condensed, we illustrate the full workflow showing how to reduce the processing time through iterative cropping/masking/filtering to narrow the study area as the analysis progresses.

We obtained the georeferenced fish occurrences from the Global Biodiversity Information Facility (GBIF; GBIF.org, 2023), the georeferenced locations of the existing dams from the Global Reservoir and Dam Database (GRanD; Lehner et al., 2011) and of the future dams (planned and under construction dams) from the Future Hydropower Reservoir and Dams Database (FHReD; Zarfl et al., 2015). We derived the raster and vector layers regarding the drainage basins, sub-catchments, stream segments, flow accumulation and flow direction from Hydrography90m.

We used the fish occurrence coordinates as input in the `get_tile_id` function to identify the IDs of the three regular 20°×20° tiles, and downloaded the Hydrography90m layers using the `download_tiles` function. We then used `crop_to_extent` to crop the rasters, `read_geopackage` to filter the vector files and `merge_tiles` to merge all layers. In a first step, we filtered the

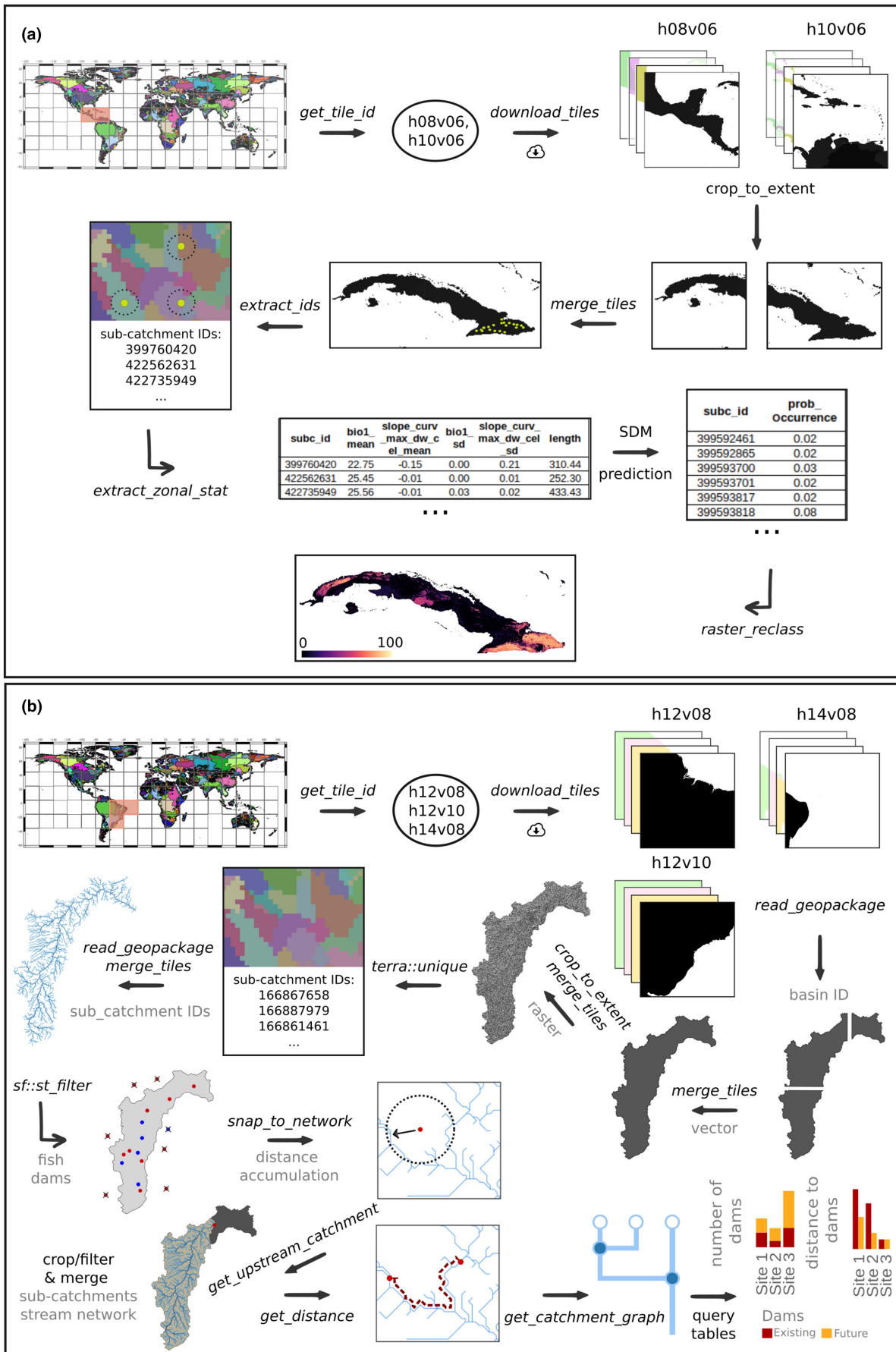


FIGURE 1 Implementation of the `hydrographr` functions in (a) a species distribution modelling analysis of a dragonfly in Cuba and in (b) a connectivity analysis workflow estimating the free-flowing river lengths between fish occurrences and dam locations in Brazil.

GeoPackage tiles of the basins for the São Francisco drainage basin and merged the tiles. We then cropped all raster layers, where we used the filtered (i.e. clipped) vector polygon (GeoPackage) of the São Francisco basin as a cutline. To identify all sub-catchment IDs in the study area, we ran the `unique` function from the `terra` package on the sub-catchment raster layer. This was done to reduce the data size and subsequent processing time, and to merge them to one single file.

To select those fish occurrences and dams that are located within the polygon of the São Francisco basin, we used the `sf_filter` function from the `sf` package. We used the `snap_to_network` function to snap the fish occurrences and dams to the stream network segments. This step was required as the spatial location of the recorded GPS points and the modelled stream network do

not necessarily match. We restricted our analyses to the upstream area of the Sobradinho dam, as this represents the known range of the species (IUCN, 2022), and delineated the upstream catchment using the `get_upstream_catchment` function. Afterward, all necessary layers and point locations were cropped/filtered to the extent of the upstream catchment. For the evaluation of the shortest distance between each fish occurrence and the closest up- and downstream dam, we first calculated the distances along the stream network between all point locations using the `get_distance` function. For any given fish occurrence, we needed to know which existing and future dam is located up- or downstream. We thus used the `get_catchment_graph` function to get all sub-catchment IDs of the stream segments up- and downstream of each fish occurrence, and then identified those stream segments with a dam. By counting

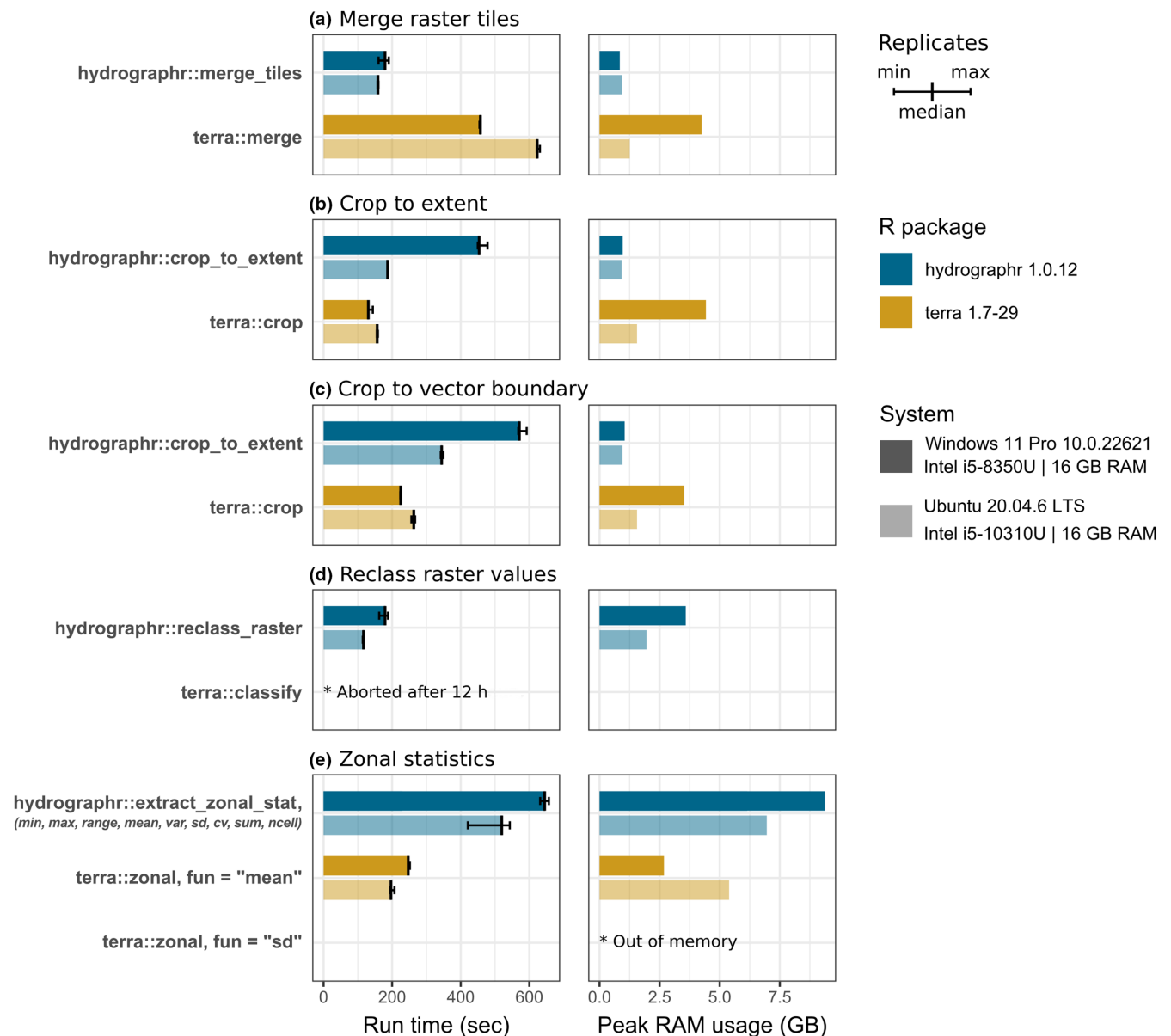


FIGURE 2 Five benchmark experiments (a–e) for selected geospatial tasks employing `hydrographr` functions (blue) and corresponding `terra` workflows (yellow). We performed each experiment three times (black bars) on a Windows (dark) and an Ubuntu (light) system and recorded the total run time (left) and the peak RAM usage (right).

the number of stream segments with a dam, we got the number of dams up- and downstream of each fish occurrence. To evaluate the shortest distance to the next dam up- and downstream of each fish occurrence, we queried the table of the network distances with our up- and downstream segments with a dam by the sub-catchment ID and identified the minimum distance.

4 | FUNCTION BENCHMARKS

We conducted benchmark experiments to compare the performance of `hydrographr` to the `terra` package (Hijmans 2023). We selected five common geospatial tasks that can be implemented with `hydrographr` and `terra` (see Figure 2 and Supporting Information S5. Benchmark):

- Merging multiple raster layers.
- Cropping a raster layer to a bounding box.
- Cropping a raster layer to an irregular vector polygon boundary.
- Reclassifying the values of a raster layer.
- Zonal statistics of a raster layer.

We performed all experiments for the Amazon river basin, which covers six Hydrography90m tiles. We recorded the total run-time and the peak RAM usage for reading, processing and writing the data and performed all experiments on a Windows and an Ubuntu system. The benchmark is documented in detail here: <https://glowabio.github.io/hydrographr/articles/benchmark.html>.

The benchmark revealed that (i) overall run-times with `hydrographr` and `terra` are comparable, (ii) `hydrographr` performed all tasks more RAM efficient than `terra`, (iii) `hydrographr` performs better on Ubuntu than on Windows, (iv) the raster reclassification and the zonal standard deviation computation failed with `terra` due to poor run-time and memory overflow while `hydrographr` successfully completed these tasks.

5 | CONCLUSIONS

`hydrographr` aims to fill the gap in freshwater geospatial data processing by combining the strengths and scalability of available open-source software with the R-interface that has become the common standard programming language in research. The currently available 21 functions provide the tools to create reproducible workflows for the first and second case studies that ran in 4 and 110min, capitalizing on 7 and 18GB of data, respectively, while not succumbing the R-session on a standard 16GB RAM laptop. This highlights that the `hydrographr` functions handle large amounts of data while users are not required to interact with the underlying software, avoiding a possible steep learning curve in learning GIS programming. Where applicable, `hydrographr` allows parallelising the functions such that the potential of `hydrographr` can be further increased with a multi-thread processing of very large datasets.

We acknowledge that geospatial data analyses and some `hydrographr` functionalities are also supported by other R packages. However, the `hydrographr` functions can be considered complementary: as most of the data processing and data extraction is performed with GDAL/OGR or GRASS GIS outside R, there is no need to import/export data into or out of R. An additional advantage of `hydrographr` is that the functions are tailored toward the Hydrography90m data that provides the most-detailed global, high-resolution hydrographic network representation to date. Therefore, the terminology and options (i.e. flags) of the functions are standardised for this data to support the analysis workflows of freshwater researchers. Still, the functions can be used also with other hydrographic datasets or even for other, general geospatial processing purposes. In this regard, we expect the package to be equally useful beyond ecological studies in the freshwater realm. Studies targeting large spatial extents and simultaneously high spatial resolution, for instance to study carbon (Ludwig et al., 2023; Raymond et al., 2013) or nutrient cycles in inland waters (Savchuk et al., 2021), can capitalise on efficient network and basin data processing. Likewise, even at smaller spatial extents, but temporally high-resolved, for example water quality/quantity data at specific intervals over long time frames, may result in equal amounts of data that require efficient processing tools.

`hydrographr` requires installing additional software, yet this one-time installation, merely copy-pasting code from the installation manual into the (WSL) console, allows bringing scalability into hydrographic analyses to users with a standard computer, which otherwise would not be possible. We intend to further develop and optimise the package toward user requirements that can be addressed at <https://github.com/glowabio/hydrographr/issues>. We aim to extend the functionality also toward lakes to integrate lake polygons into the network, or to delineate lake catchments. We also endeavour to further tap on the functionalities offered by network graphs.

AUTHOR CONTRIBUTIONS

Sami Domisch contributed to the conception, and Marlene Schürz and Afroditi Grigoropoulou developed and maintained the package. Marlene Schürz, Afroditi Grigoropoulou, Jaime García Márquez, Yusdiel Torres-Cambas, Thomas Tomiczek and Sami Domisch wrote the functions and manual descriptions. Vanessa Bremerich and Giuseppe Amatulli provided support in GDAL/OGR, GRASS GIS, and AWK during the function development. Vanessa Bremerich contributed to testing and debugging. Marlene Schürz and Afroditi Grigoropoulou adapted the functions for Windows and Linux, and built the functions and manage the GitHub repository. Mathieu Flourey, Afroditi Grigoropoulou and Marlene Schürz tested and edited functions. Christoph Schürz and Marlene Schürz performed the benchmarking. Marlene Schürz and Afroditi Grigoropoulou developed the case studies, with contributions from all the authors. Marlene Schürz, Afroditi Grigoropoulou and Sami Domisch developed the website of the R-package. Marlene Schürz and Afroditi Grigoropoulou drafted the manuscript and the figures. Sami Domisch, Jaime García Márquez, Yusdiel Torres-Cambas, Thomas Tomiczek,

Mathieu Flourey, Christoph Schürz, Giuseppe Amatulli, Vanessa Bremerich and Hans-Peter Grossart reviewed and commented on the manuscript. All authors read and approved the final version of the manuscript. Marlene Schürz and Afroditi Grigoropoulou contributed equally to the package and manuscript.

ACKNOWLEDGEMENTS

We thank NFDI4Biodiversity for providing the funding that allowed us to develop the `hydrographr` R-package by integrating Bash scripts used in geospatial workflows into R functions. The partners of the NFDI4Biodiversity consortium are funded by the German Research Foundation (DFG) within the framework of the agreement between the Federal Government and the Länder on the establishment and funding of the National Research Data Infrastructure (NFDI) of 26 November 2018 (DFG project number 442032008). In addition, this work has been funded by the German Research Foundation (NFDI4Earth, DFG project no. 460036893, <https://www.nfdi4earth.de/>). Moreover, we acknowledge funding by the Leibniz Competition awarded to Sami Domisch (J45/2018) and by the German Federal Ministry of Education and Research (BMBF grant agreement number no. 033W034A) and by the Alexander von Humboldt Foundation awarded to Yusdiel Torres-Cambas (Ref 3.2-CUB-1212347-GF-P).

CONFLICT OF INTEREST STATEMENT

There are no conflicts of interest.

PEER REVIEW

The peer review history for this article is available at <https://www.webofscience.com/api/gateway/wos/peer-review/10.1111/2041-210X.14226>.

DATA AVAILABILITY STATEMENT

The R-package is available in the GitHub repository <https://github.com/glowabio/hydrographr> (Schürz et al., 2023). The hydrographic data used in the case studies are stored in <https://glowabio.org/project/hydrography90m/>, the species data in <https://doi.org/10.18728/igb-fred-778.3>, <https://doi.org/10.15468/8cxijb> and <https://doi.org/10.15468/dl.hpt8k8>, the climatic data in https://envicloud.wsl.ch/#/?prefix=chelsa%2Fchelsa_V2%2FGLOBAL%2F and the dam locations in <https://www.globaldamwatch.org/directory>. All data can be downloaded and processed using the code of the case studies, stored in https://glowabio.github.io/hydrographr/articles/case_study_cuba.html and https://glowabio.github.io/hydrographr/articles/case_study_brazil.html.

ORCID

Afroditi Grigoropoulou  <https://orcid.org/0000-0002-7884-097X>

Jaime García Márquez  <https://orcid.org/0000-0002-1998-5669>

Yusdiel Torres-Cambas  <https://orcid.org/0000-0003-2312-2329>

Mathieu Flourey  <https://orcid.org/0000-0002-4952-5807>

Vanessa Bremerich  <https://orcid.org/0000-0002-7657-1534>

Christoph Schürz  <https://orcid.org/0000-0002-7204-5828>

Giuseppe Amatulli  <https://orcid.org/0000-0002-8341-2830>

Hans-Peter Grossart  <https://orcid.org/0000-0002-9141-0325>

Sami Domisch  <https://orcid.org/0000-0002-8127-9335>

REFERENCES

- Aho, A. V., Kernighan, B. W., & Weinberger, P. J. (1979). Awk—A pattern scanning and processing language. *Software: Practice and Experience*, 9, 267–279.
- Altermatt, F. (2013). Diversity in riverine metacommunities: A network perspective. *Aquatic Ecology*, 47, 365–377.
- Amatulli, G., Casalegno, S., D'Annunzio, R., Haapanen, R., Kempeneers, P., Lindquist, E., Pekkarinen, A., Wilson, A., & Zurita-Milla, R. (2014). Teaching spatiotemporal analysis and efficient data processing in open source environment. In *Proceedings of the 3rd Open Source Geospatial Research & Education Symposium* (p. 13). <https://api.semanticscholar.org/CorpusID:64709216>
- Amatulli, G., Garcia Marquez, J., Sethi, T., Kiesel, J., Grigoropoulou, A., Üblacker, M. M., Shen, L. Q., & Domisch, S. (2022). Hydrography90m: A new high-resolution global hydrographic dataset. *Earth System Science Data*, 14, 4525–4550.
- Domisch, S., Friedrichs, M., Hein, T., Borgwardt, F., Wetzig, A., Jähnig, S. C., & Langhans, S. D. (2019). Spatially explicit species distribution models: A missed opportunity in conservation planning? *Diversity and Distributions*, 25, 758–769.
- Domisch, S., Jaehrig, S. C., Simaika, J. P., Kummerlen, M., & Stoll, S. (2015). Application of species distribution models in stream ecosystems: The challenges of spatial and temporal scale, environmental predictors and species occurrence data. *Fundamental and Applied Limnology*, 186, 45–61.
- Fotheringham, A., & Wong, D. (1991). The modifiable areal unit problem in multivariate statistical analysis. *Environment and Planning A: Economy and Space*, 23, 1025–1044.
- GBIF.org. (2023). GBIF occurrence download. <https://doi.org/10.15468/dl.hpt8k8>
- GDAL Development Team. (2020). GDAL-geospatial data abstraction library, version 3.1.0. Open Source Geospatial Foundation.
- GNU Project. (2007). Free software foundation. Bash (3.2. 48)[unix shell program].
- GRASS Development Team. (2022). *Geographic resources analysis support system (GRASS GIS) software, version 8.2*. Open Source Geospatial Foundation.
- Hijmans, R. J. (2023). *terra: Spatial data analysis*. R package version 1.7-3.
- IUCN. (2022). The IUCN red list of threatened species. <https://www.iucnredlist.org>
- Jähnig, S. C., Baranov, V., Altermatt, F., Cranston, P., Friedrichs-Manthey, M., Geist, J., He, F., Heino, J., Hering, D., Hölker, F., Jourdan, J., Kalinkat, G., Kiesel, J., Leese, F., Maasri, A., Monaghan, M. T., Schäfer, R. B., Tockner, K., Tonkin, J. D., & Domisch, S. (2021). Revisiting global trends in freshwater insect biodiversity. *WIREs Water*, 8, e1506.
- Karger, D. N., & Zimmermann, N. E. (2019). *Climatologies at high resolution for the earth land surface areas chelsa v1. 2: Technical specification*. Swiss Federal Research Institute WSL.
- Lai, J., Lortie, C. J., Muenchen, R. A., Yang, J., & Ma, K. (2019). Evaluating the popularity of r in ecology. *Ecosphere*, 10, e02567.
- Lehner, B., Liermann, C. R., Revenga, C., Vörösmarty, C., Fekete, B., Crouzet, P., Döll, P., Endejan, M., Frenken, K., Magome, J., Nilsson, C., Robertson, J. C., Rödel, R., Sindorf, N., & Wisser, D. (2011). High-resolution mapping of the world's reservoirs and dams for sustainable river-flow management. *Frontiers in Ecology and the Environment*, 9, 494–502.
- Ludwig, S. M., Natali, S. M., Schade, J. D., Powell, M., Fiske, G., Schiferl, L., & Commane, R. (2023). Scaling waterbody carbon dioxide and methane fluxes in the arctic using an integrated terrestrial-aquatic approach. *Environmental Research Letters*, 18, 064019.

- Meyer, C., Kreft, H., Guralnick, R., & Jetz, W. (2015). Global priorities for an effective information basis of biodiversity distributions. *Nature Communications*, 6, 1–8.
- Moore, I. D., Grayson, R., & Ladson, A. (1991). Digital terrain modelling: A review of hydrological, geomorphological, and biological applications. *Hydrological Processes*, 5, 3–30.
- Pebesma, E. (2018). Simple features for R: Standardized support for spatial vector data. *The R Journal*, 10, 439–446.
- Pizzini, K., Bonzini, P., Meyering, J., & Gordon, A. (2018). Gnu sed, a stream editor. *Abgerufen am*, 7.
- QGIS Development Team. (2021). *QGIS geographic information system*. QGIS Association.
- R Core Team. (2022). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing.
- Raymond, P. A., Hartmann, J., Lauerwald, R., Sobek, S., McDonald, C., Hoover, M., Butman, D., Striegl, R., Mayorga, E., Humborg, C., Kortelainen, P., Dürr, H., Meybeck, M., Ciais, P., & Guth, P. (2013). Global carbon dioxide emissions from inland waters. *Nature*, 503, 355–359.
- Redlands, C. E. S. R. I. (2011). Arcgis desktop: Release 10.
- Reid, A. J., Carlson, A. K., Creed, I. F., Eliason, E. J., Gell, P. A., Johnson, P. T. J., Kidd, K. A., MacCormack, T. J., Olden, J. D., Ormerod, S. J., Smol, J. P., Taylor, W. W., Tockner, K., Vermaire, J. C., Dudgeon, D., & Cooke, S. J. (2019). Emerging threats and persistent conservation challenges for freshwater biodiversity. *Biological Reviews*, 94, 849–873.
- Savchuk, O. P., Isaev, A. V., & Filatov, N. N. (2021). Modeling of the large-scale nutrient biogeochemical cycles in Lake Onego. *Biogeosciences Discussions*, 2021, 1–26.
- Schürz, M., Grigoropoulou, A., Marquez, J. G., Torres-Cambas, Y., Tomiczek, T., Flourey, M., Bremerich, V., Schürz, C., Amatulli, G., & Domisch, S. (2023). glowabio/hydrographr: Hydrographr 1.0.16 (v1.0.16). *Zenodo* <https://doi.org/10.5281/zenodo.8355813>
- Tange, O. (2011). Gnu parallel—the command-line power tool. *The USENIX Magazine*, 36, 42–47.
- Torres-Cambas, Y., & Salazar Salina, J. C. (2022). Ephemeroptera, trichoptera and odonata of cuba. <https://doi.org/10.18728/igb-fred-778.3>
- Valavi, R., Elith, J., Lahoz-Monfort, J. J., & Guillera-Arroita, G. (2021). Modelling species presence-only data with random forests. *Ecography*, 44, 1731–1742.
- van Klink, R., Bowler, D. E., Gongalsky, K. B., Swengel, A. B., Gentile, A., & Chase, J. M. (2020). Meta-analysis reveals declines in terrestrial but increases in freshwater insect abundances. *Science*, 368, 417–420.
- Ver Hoef, J., Peterson, E., Clifford, D., & Shah, R. (2014). Ssn: An r package for spatial statistical modeling on stream networks. *Journal of Statistical Software*, 56, 1–45.
- Verdin, K., & Verdin, J. (1999). A topological system for delineation and codification of the earth's river basins. *Journal of Hydrology*, 218, 1–12.
- Wright, M. N., & Ziegler, A. (2017). Ranger: A fast implementation of random forests for high dimensional data in C++ and R. *Journal of Statistical Software*, 77, 1–17.
- Zarfl, C., Lumsdon, A. E., Berlekamp, J., Tydecks, L., & Tockner, K. (2015). A global boom in hydropower dam construction. *Aquatic Sciences*, 77, 161–170.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

Supporting information S1–S5.

How to cite this article: Schürz, M., Grigoropoulou, A., García Márquez, J., Torres-Cambas, Y., Tomiczek, T., Flourey, M., Bremerich, V., Schürz, C., Amatulli, G., Grossart, H.-P., & Domisch, S. (2023). hydrographr: An R package for scalable hydrographic data processing. *Methods in Ecology and Evolution*, 14, 2953–2963. <https://doi.org/10.1111/2041-210X.14226>