scientific reports

OPEN

Reproducible Tract Profiles 2 (RTP2) suite, from diffusion MRI acquisition to clinical practice and research

Garikoitz Lerma-Usabiaga [1,2,3,4 ✉], Mengxing Liu [2], Pedro M. Paz-Alonso [2,4] & Brian A. Wandell [1,3]

Diffusion MRI is a complex technique, where new discoveries and implementations occur at a fast pace. The expertise needed for data analyses and accurate and reproducible results is increasingly demanding and requires multidisciplinary collaborations. In the present work we introduce Reproducible Tract Profiles 2 (RTP2), a set of flexible and automated methods to analyze anatomical MRI and diffusion weighted imaging (DWI) data for reproducible tractography. RTP2 reads structural MRI data and processes them through a succession of serialized containerized analyses. We describe the DWI algorithms used to identify white-matter tracts and their summary metrics, the flexible architecture of the platform, and the tools to programmatically access and control the computations. The combination of these three components provides an easy-to-use automatized tool developed and tested over 20 years, to obtain usable and reliable state-of-the-art diffusion metrics at the individual and group levels for basic research and clinical practice.

Due to the complexity of neuroimaging data and analyses it is challenging for researchers to implement reproducible research methods. In many neuroimaging publications, it is impossible for readers to replicate the experimental data acquisition or reproduce the computational analyses[1–3]. Current practices increasingly invite data sharing among researchers and labs to enable colleagues in the scientific community and clinical practitioners to reproduce, check, and further explore datasets and computational analyses[4,5]. Together with the desirable increase of good practices and data sharing there has been a substantial increase in algorithm complexity. Some of this complexity takes the form of a large number of pipeline parameters that users can set. Choices made regarding these parameters can have substantial effects on the reported MRI results[6,7]. Further, it has been pointed out across different MRI techniques that we are often uncertain about critical parameters in computational models[8]. The two techniques focused on in this paper, DWI and tractography, are no exception. It is difficult for neuroscientists to keep track of the full range of parameters used in any particular analysis, and few neuroscientists keep records of the combinations of parameters they used during data exploration and exploitation[9]. These difficulties can be exacerbated in clinical settings, where clinicians may have access to a large cohort of patients and less time to develop computational research protocols. Here, we introduce a new tool, Reproducible Tract Profiles 2 (RTP2), that can simplify the execution of reproducible tractography analyses for non-experts.

Other solutions are available for identifying white matter tracts. Those solutions vary greatly in terms of scope, technical approach, and flexibility. The main differences are based on the level at which each solution operates. For example, some, such as MRtrix[10], FSL[11], Tracula[12], TrackVis[13], DSI-studio[14] or Dipy[15], are complete ecosystems where most of the processing steps (preprocessing, registering, modeling and tracking) are included. Using any of these solutions, the researcher is able to obtain the tractography results, and build a pipeline on top of them to automatize most of the tasks. On the other hand, there are some meta-solutions that are built on top of existing neuroimaging tools, relying on the best available options for image registration, preprocessing, anatomical analyses, diffusion modeling, tractometry, and so forth. In this category, we include TractoFlow[16] and TractSeg[17], as well as both the Matlab and Python versions of AFQ[18,19]. RTP2 falls into this second category. In RTP2 we started out with a clear set of objectives that could overcome the lack of functionalities we suffered with

[1]Department of Psychology, Stanford University, 450 Serra Mall, Jordan Hall Building, Stanford, CA 94305, USA. [2]BCBL, Basque Center on Cognition, Brain and Language, Mikeletegi Pasealekua 69, 20009 Donostia-San Sebastián, Gipuzkoa, Spain. [3]Wu Tsai Neurosciences Institute, Stanford University, Stanford, CA 94305, USA. [4]IKERBASQUE, Basque Foundation for Science, 48013 Bilbao, Spain. ✉email: garikoitz@gmail.com

1

the first version of RTP[20,21]. RTP was based on a containerization of the Matlab-based tool AFQ. First, we selected our target user population as non-diffusion experts such as cognitive neuroscientists, neuroanatomists, machine learning practitioners, or clinicians who are interested in locating a set of tracts in a reproducible, automated, flexible and robust manner (see[22] for an example application). Therefore, we knew that the solution needed to be complete: from the acquisition of images at the scanner and all the way to the tracts and their metrics. Second, we wanted to maintain the original objective of obtaining a fully reproducible pipeline guaranteeing data provenance. We took a holistic approach and although we tried to maintain as much as possible from the original codebase, as it has been shown to be robust and reliable, in RTP2 we removed most of the Matlab base code and we relied more on new code that manages Freesurfer[23], MRtrix[10] and other well-established open source software tools (see Methods section for details). On top of that, we implemented a new ROI definition container, integrating several tools and atlases, as well as a new preprocessing container, which was included as part of AFQ/RTP before. The whole result taken together goes way beyond the original RTP tool. RTP2 is an automated tool that builds on the state-of-the-art computations of DWI data and tractography and improves computational reproducibility. This solution is reproducible because the analysis software and its dependencies are embedded in a container, and the input data, complete set of analysis, configuration parameters and outputs are stored in a single informatics platform. This platform can be hosted on a single dedicated computer or a neuroinformatics platform that runs locally, on a server, or in the cloud.

RTP2 has been designed with three objectives: To be (1) reproducible, (2) automated, and (3) flexible. The system ensures *reproducibility* by basing the computations on containers that include the software and all of its dependencies, and that define the parameters using configuration files that are stored as part of the results. The basic steps are laid out in Fig. 1. A container (HeuDiConv) reads the DICOMs from the scanner, converts them to Nifti and curates and transforms them in BIDS format[24]. Next, the anatROIs container processes the T1-weighted images using Freesurfer. The researcher can specify ROIs in normalized space, using existing ones in RTP2 or creating project-specific ROIs in MNI or fsaverage space, that will be transformed into the individual-subject space. If desired, some ROIs can be added in the individual-subject native space that will be directly passed to the next steps without transformation. The RTP2-preproc container processes the diffusion images
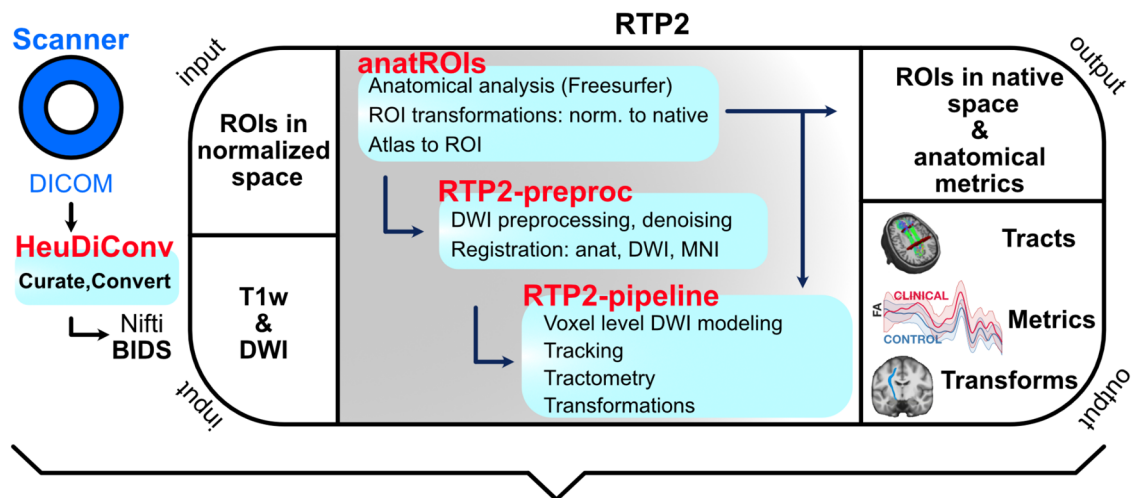
Figure 1. Automated and reproducible pipeline for tractography. RTP2 can be automated to produce reproducible results with minimal human interaction. The DICOMs acquired from the MRI scanner are automatically converted into BIDS format Nifti-s and transferred to the informatics platform. Containerized images containing the anatomical and DWI analysis tools along with files specifying the computational parameters are stored and managed in the system. The system takes individual-subject space anatomical images, DWI data and ROIs in normalized space that researchers would like to use in subsequent steps and first coregisters them. Then it runs the anatomical pipeline and creates structural segmentations and ROIs with Freesurfer, as well as multiple anatomical metrics. The system converts multiple public atlases as well as MNI/ surface defined ROIs into volumetric ROIs to be used later in tractography. After the DWI preprocessing, the user can select any combinations of ROIs to define white matter tracts. Tracts can be defined using seed and target ROIs, or can be selected from all the fibers of a whole-brain tractogram (WBT). In the output of the pipeline the user will obtain all the requested tracts and metrics, including along-the-tract profiles (see Fig. 5). All structural metrics from Freesurfer as well as all the ROIs in individual-subject space generated from the atlases or input by the researcher will be available. Those ROIs can be used for other analyses, such as fMRI. The whole system can be installed once on a single computer, installed in a distributed system working with high performance computing clusters, or managed in a cloud-based neuroinformatic platform.

and registers them with the anatomical images. The RTP2-pipeline container combines the anatomical images, the preprocessed diffusion images and the ROIs to obtain white matter tracts of interest. This container outputs data in a number of ways: tracts in Nifti and tck formats, tracts in volumetric Nifti-s in individual and template space, tract metrics in csv format, and more. The four containers can be installed on a single computer, in a High-Performance Computing (HPC) cluster or in a neuroinformatics platform.

RTP2 is highly *automated* because it is designed for cognitive neuroscientists and/or clinicians interested in conducting research on white matter and tractography. The system quantifies the properties of white-matter tracts at the individual-subject level. It also offers the user high flexibility in terms of how to define white-matter tracts based on white-matter, subcortical and cortical ROIs, as well as the most representative tractometrics, such as fractional anisotropy (FA) or radial diffusivity (RD)[25], and tract-similarity metrics (e.g., Dice). The goal is to provide multiple options for defining the input terms to make the system as easy to use as possible. For the advanced user, this system allows for systematic and programmatic testing of a large set of parameters, with the objective of optimizing the final results. The architecture of the solution, the defaults and the algorithms are explained in the Methods section and in the wiki of the tool (https://garikoitz.github.io/RTP-pipeline/).

Flexibility is provided in two ways: (i) the system can be run in a variety of platforms, depending on the resources available in the lab/clinic; the results will be the same across platforms, and (ii) the user can decide the diffusion model to use, the tracking algorithm, and what to track—with the pre-installed and user created ROIs, the possibilities are practically infinite. In Fig. 2 we illustrate all the possible different ROIs RTP2 can use to define tracts. We provide an example of a representative tract to illustrate how those ROIs can be used. The ROIs can be classical anatomically defined white-matter tracts (Fig. 2A), which can be used as targets and seeds
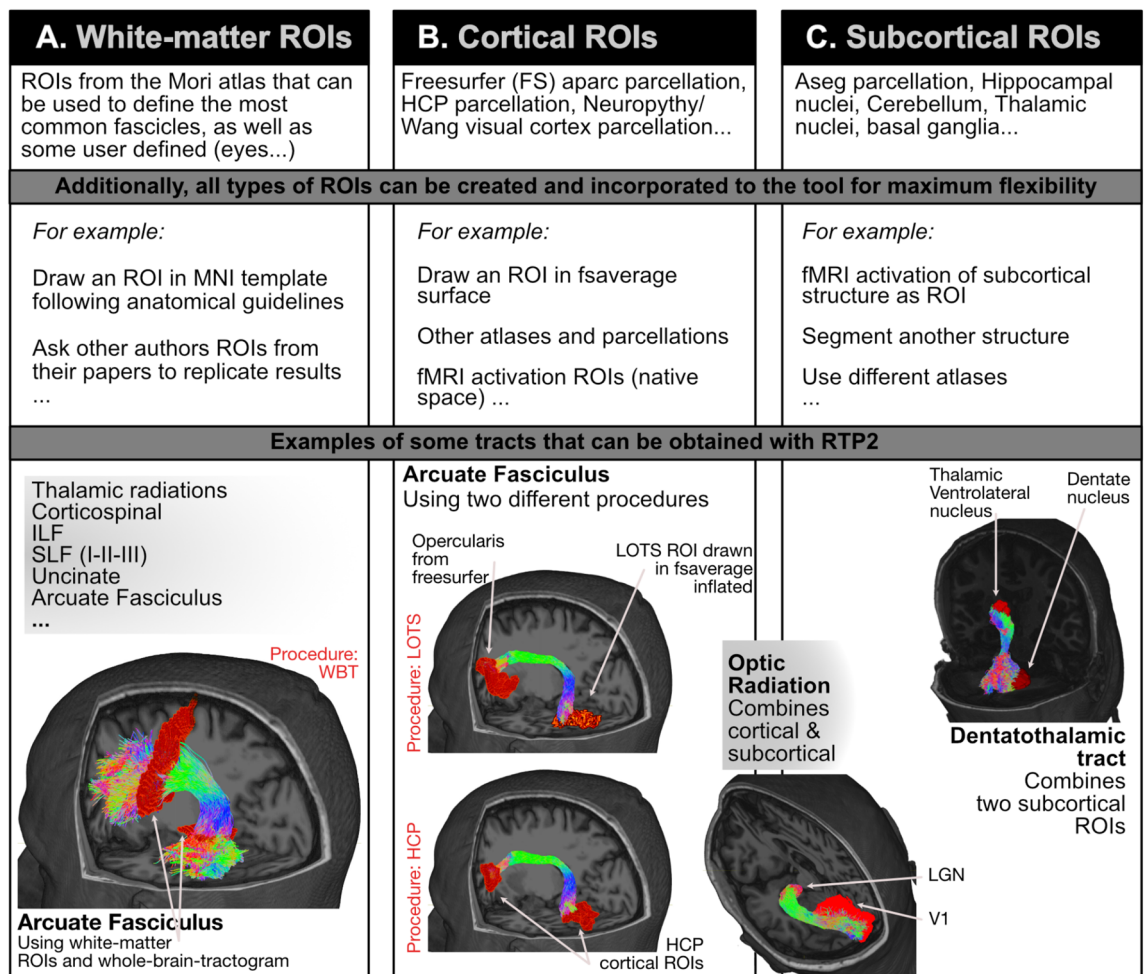

Figure 2. RTP2 Flexibility: any combination of ROIs and several diffusion models can be used to measure tracts of interest. The RTP2 tool allows for an almost infinite combination of ROIs to obtain tracts. These ROIs can be classical anatomically defined white-matter tracts (see examples in first row), which can be used to track in any combination: (**A**) ROIs are used to select fibers from a whole-brain tractogram, (**B**) ROIs are cortical structures, and (**C**) subcortical ROIs. Additionally, the system is prepared to input user defined ROIs without any coding (see examples in second row). We include some example tracts that can be obtained from the pre-installed ROIs to illustrate how RTP2 tracking works, but the system is fully configurable by the researcher. In red examples for the three procedures used to illustrate the reproducibility analysis. WBT: Whole-brain tractography, HCP: Human connectome project, LOTS: Lateral occipito-temporal sulcus.

themselves, or can be used to select fibers from a whole-brain tractogram. The ROIs can also be cortical (Fig. 2B) or subcortical structures (Fig. 2C). All types of ROIs can be combined to obtain any number of tracts. In the first row of Fig. 2 we list most of the ROIs already included in RTP2, based on different atlases and tools. Importantly, the system is prepared to input user defined ROIs without any coding; in the second row of Fig. 2, we include examples of ROIs that can be created by the user. In the third row we include some example tracts. In Fig. 2A we illustrate the white matter ROIs tracking the arcuate fasciculus: first we obtained the whole-brain tractogram, and then used two white matter ROIs to just select those fibers crossing both ROIs. In Fig. 2B we show the arcuate fasciculus tracked using cortical ROIs only, using both ROIs bidirectionally as targets and seeds. Next, to illustrate how we can combine subcortical and cortical ROIs, between Fig. 2B,C we show the optic radiation, tracked using the same bidirectional target-seed method. Finally, we use the same method with two subcortical ROIs (Fig. 2C) to obtain the dentatothalamic tract.

Researchers have different preferences and priorities regarding how to track different tracts. A comparison of the three different procedures to track the arcuate fasciculus shown in Fig. 2 illustrates the potential and flexibility of RTP2 to enable the implementation of different approaches and choices. It also shows the differences such tracking choices might lead to in the data obtained. Moreover, the analytical approach used here allows a comparison of the computational and test–retest reproducibility of the three different procedures. In the Methods section at the end of this work, we describe the computational steps involved, and some of the multiple possible configurations allowed.

The first procedure is based on an 8 million fiber whole-brain tractography (i.e., WBT; marked in red in Fig. 2A): we select the fibers that cross two white-matter ROIs to define the arcuate fasciculus. In the second procedure, we use the Human Connectome Project (HCP; in red in Fig. 2B) cortical atlas[26] to select one ROI corresponding to the opercularis region in the inferior-frontal gyrus (IFG) and a second ROI that corresponds to the occipito-temporal sulcus (OTS). We use both ROIs as seeds and targets for tracking the arcuate fasciculus. In the third procedure to track the AF we obtain the ROIs differently: the opercularis region is obtained from Freesurfer's aparc2009 cortical parcellation. The ventral occipito-temporal region ROI is the lateral OTS (LOTS) manually drawn on the inflated surface of Freesurfer's fsaverage (procedure referred to as LOTS in red in Fig. 2B). The manually drawn surface ROI is automatically converted to the individual-subject volumetric ROI, in a step included in RTP2. For each of the three differently tracked arcuate fasciculus, we obtained the profile of FA values along the tract (see Fig. 5), as well as the mean value of the whole tract, and compared the values of the three differently tracked arcuate fasciculus. We obtain very high computational and test–retest reproducibility in most cases. This example illustrates how researchers can flexibly control and compare the multiple solutions that can be implemented with this tool according to their research questions and objectives. It also demonstrates how tracking choices impact the final results obtained.

Results

To illustrate the flexibility and reproducibility of RTP2, we analyzed the bilateral arcuate fasciculus (AF) from a dataset of 112 participants (24 repeated for test–retest) using the three above-mentioned procedures: (1) WBT: using a whole-brain-tractogram and white-matter ROIs—Fig. 2A, (2) HCP: using cortical ROIs obtained from the HCP atlas—Fig. 2C, and (3) LOTS: using cortical ROIs obtained from Freesurfer's atlas and hand-drawn in Freesurfer's average surface—Fig. 2C. We used a double analytical approach: *computational reproducibility,* repeating the computation on the same diffusion data and quantifying changes from computation to computation; and *test–retest reproducibility,* obtaining DWI data from the same subjects and using the same MRI protocol in two different sessions. As an initial illustration of the analysis, in Table 1 we show the correlations of the tract's mean FA values for computational and test–retest measures for all three procedures.

Next, we provide computational and test–retest reproducibility analyses based on the whole tract. We show the FA values in the text and other measures in Supplementary Figures (Figs. S1–3).

Computational reproducibility. To test the computational reproducibility, we repeated the tractography reconstruction using the RTP2-pipeline twice using identical inputs and parameters. For each tract of interest, computational reproducibility was measured with both microstructual (i.e., FA, MD, RD) and macrostructural metrics (i.e., Dice, density correlation, bundle adjacency; Fig. 3). The generation of the tract profiles for each tract in RTP2 allows users to conduct further quantitative analysis, not restricted to the mean values of the tract. The computational reproducibility of each individual data point along the tract is good, with FA across-subject

	Computational (r, N = 112)	Test–retest (r, N = 24)
Left WBT (L_WBT)	0.990	0.962
Right WBT (R_WBT)	0.994	0.962
Left HCP (L_HCP)	0.995	0.953
Right HCP (R_HCP)	0.989	0.942
Left LOTS (L_LOTS)	0.988	0.955
Right LOTS (R_LOTS)	0.984	0.933

Table 1. Evaluation of computational and test–retest reproducibility for arcuate fasciculus mean FA values for the three methods.
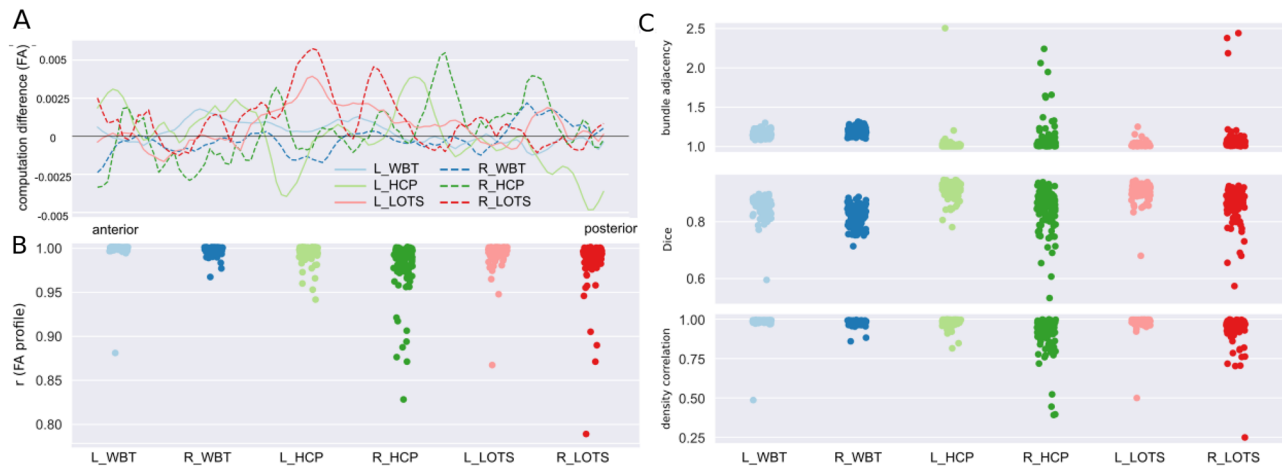
Figure 3. Evaluation of computational reproducibility for arcuate fasciculus tracking across three procedures. (**A**) Per procedure/hemisphere across-subject mean FA tract-profile differences between the first and the second computations. (**B**) Strip plots showing the distribution of correlation coefficients between all possible pairs computed for each tract and each subject. Each dot represents the correlation coefficient for a specific computation pair for one participant. (**C**) Macrostructural agreement indices distribution: bundle adjacency (top), Dice coefficient (middle), and density correlation (bottom) between the two separate computations for each tract and each subject.

average difference between computations being smaller than 0.005 (see Fig. 3A). The correlation between the tract profiles for computation 1 and computation 2 tends to be high as well, with the average value in the range of 0.978 ± 0.03 for R_HCP to 0.997 ± 0.01 for R_WBT (Fig. 3B).

The R_HCP and R_LOTS procedures yielded median correlations across all subjects of 0.987 and 0.992, respectively, although the data from some subjects showed relatively lower correlation values, with correlations ranging from 0.828 to 0.999 and 0.789 to 0.999 at the subject level. At the macrostructural level, we used several measures to describe reproducibility of streamlines (Fig. 3C). This includes (1) volume Dice overlap, which describes the overlap of two tracts, (2) density correlation which describes the similarity of the streamline density of two tracts, (3) bundle adjacency which describes the average distance of disagreement of two tracts. The average Dice overlap ranges from 0.82 ± 0.04 to 0.91 ± 0.03 across the tracts of interest, which reflects that all these tracts have high volume overlap between two separate computations. This high computational reproducibility observed for Dice overlap is also found for the other macrostructural indexes (i.e., density correlation and bundle adjacency) across all the tracts. However, similar to the FA value correlations, the R_HCP and R_LOTS procedures showed high variance for the macrostructural measurements, with density correlation values ranging from 0.39 to 0.99 and from 0.25 to 0.99 at the subject level.

Test–retest reproducibility. The test–retest reproducibility was assessed on a subset of 24 participants, who returned for a second acquisition session within a mean temporal interval of 15 days. The same analytical approaches were adopted to measure test–retest reproducibility using microstructural and macrostructural measurements. Overall, we found high test–retest reproducibility across all the left and right hemisphere white-matter, subcortical and cortical tracts of interest, although some specific tracts showed numerically higher variability. As expected, the reproducibility values are lower in test–retest than in computational reproducibility, with overall higher mean difference between tract profiles in test–retest reproducibility (see Fig. 4A). This affects the FA correlation analysis, as well as the coefficients averaged within each tract, which ranged from 0.918 ± 0.11 for R_HCP to 0.990 ± 0.01 for L_WBT (see Fig. 4B). Again, R_HCP and R_LOTS showed higher numerical variability between test and retest computations, with the subject level FA correlation values for R_HCP ranging from 0.46 to 0.99, and from 0.76 to 0.99 for R_LOTS. At the macrostructural level, the average Dice overlap has a range of 0.78 ± 0.04 to 0.89 ± 0.03 across all the tracts of interest. Consistent with the Dice overlap results, all tracts have high agreement in streamline density and small bundle adjacency. Similar results were observed in terms of variance, with R_HCP and R_LOTS showing numerically higher variance in all indices.

Discussion

The increasing complexity in diffusion neuroimaging data and analyses challenges researchers' ability to implement reproducible research methods. The technical and conceptual expertise required to obtain reproducible results is being addressed in the neuroimaging community through the creation of automated tools that enable non-experts to conduct analyses using state-of-the-art methods with better validity and reliability. Additionally, some of the complexity takes the form of the large number of parameters that users can set. Choices made in these parameters can have substantial effects on the reported results, and it is critical to have a system that keeps track of the parameters used in any particular analysis and records the combinations of parameters used during data exploration[9]. With these objectives in mind, we developed a reproducible, automated and flexible system
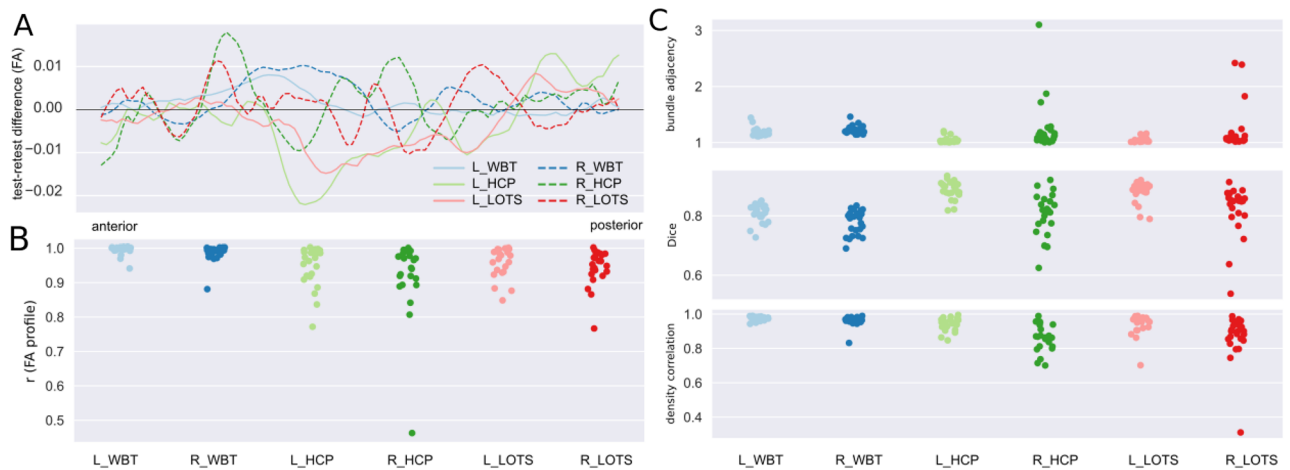
Figure 4. Evaluation of test–retest reproducibility for arcuate fasciculus tracking across three procedures. (**A**) Per procedure/hemisphere across-subject mean FA tract-profile differences between the test and retest acquisitions. (**B**) Strip plots showing the distribution of the correlation coefficients between test and retest computations for each tract and each subject. (**C**) Macrostructural agreement indices distribution: bundle adjacency (top), Dice coefficient (middle), and density correlation (bottom) for test and retest computations for each tract and each subject.

that, once installed, can automatically run the tractography pipeline, while keeping track of all the analytical steps performed.

We illustrated the functionality and the reproducibility of the pipeline by examining computational reproducibility on 112 subjects and test–retest reproducibility on 24 of those subjects with repeated measures. We obtained nearly perfect computational reproducibility values in all metrics, at the mean tract level as well as at the whole-tract level. We illustrated the flexibility of the tool by using it to compute the same tract, the AF, in three different ways. Given the same anatomical structure can be defined in several different ways 7, this is common practice in the field, as well as different laboratories using different pipelines and software tools. In our examples, all ROIs came from an average space and were converted to individual-subject space, but RTP2 also allows users to define ROIs directly in native space, e.g., as a result of individual subject fMRI experiments. Results revealed that the computational reproducibility is nearly perfect across all three procedures tested. However, as expected, it is not absolutely perfect, due to several non-deterministic calculations contained within the pipeline, all of which can slightly influence the final results. A fully reproducible pipeline can be achieved by having the random seeding initialization fixed and the interested researcher will be able to choose such models in RTP2. In our analyses we used a probabilistic algorithm to generate the streamlines, as this generally provides the best results for most tracts[27–29]. Additionally, having the random seeding fixed is not compatible with multi-threaded steps. Multi-threading allows for faster computations but introduces randomness in the order of execution and also tends to slightly impact the reproducibility of results.

As indicated, RTP2 can also be used in clinical research with clinical populations. For example, this tool can be used for surgery preparation (e.g., tracking the dentatothalamic tract in patients with Parkinson disease), and it can be used to further understand the involvement of white-matter tracts in the spreading of grand mal (e.g., patients with intractable epilepsy). RTP2 can also be used in infants with brain tumors, overcoming a triple challenge: (1) low myelination of infant brains, and therefore, weaker diffusion signals, (2) different head sizes, requiring the use of non-standard templates, and (3) non-standard tracts, as the tumors deviate standard natural pathways. Thus, RTP2 is robust enough to work with different clinical populations across the whole life span, and it is flexible enough to allow many different possibilities to be explored, even in patients with structural alterations. Since RTP2 has no embedded priors, it opens the door to model any shaped tracts at the individual subject level.

This project is open-source, it is available in Github and Dockerhub to any researcher interested in using it. This also means that anyone can extend its functionalities, making them available to every user. We made the tool as easy-to-use and as flexible as possible, so the average user should have no need to add any functionalities. This is important as one possible limitation of the tool is that any researcher wishing to add new functionalities to RTP2 will need expert knowledge of Docker containers, server configurations and bash scripting.

There are important considerations for researchers interested in using this tool or diffusion in general. For example, our results indicate that the WBT procedure, based on a whole-brain tractogram and white-matter ROIs, provided the most reproducible results. However, these results come at a cost of precision and specificity. With the WBT procedure, there is limited control over the cortical endings of the tract, and it usually only works well with well-known big fascicles. For other smaller or more complicated tracts (e.g., dentatothalamic tract, optic radiation), the subcortical-cortical or cortico-cortical approaches provide slightly better results. Depending on the tract of interest, it is always worth testing different procedures in a small number of subjects to find out which yields the best results. Additionally, we observed that as the size of the ROI decreases, so do both the computational and test–retest reproducibility. This is not specific to our tool; as above, researchers will need to

decide how to combine the best ROIs and parameters to obtain the best possible tracts in the corresponding dataset/population. We plan to add as many recommendations as possible in the RTP2 wiki, so that researchers can conduct their own analyses from an advanced starting point.

In conclusion, the RTP2 solution implements fully reproducible analyses, in an automated manner, while maintaining high levels of flexibility. The methods we used guarantee that each tract profile can be traced to the original DICOMs. RTP2 implements reproducible computational methods by embedding algorithms on containers. In all cases, RTP2 will provide the same results if the input data is the same. Most importantly, this will be the case whether the data is reanalyzed now or in several years. We believe that RTP2 can be a useful neuroimaging software tool which will help improve standards in neuroscientific research and outcomes in clinical settings. Crucially, it is consistent with modern requirements in being reproducible, automated/easy-to-use, and flexible.

Methods

Here we describe the most important characteristics of RTP2, after providing MRI data acquisition details (used in this paper as an example dataset). Detailed information, instructions about how to run RTP2, and example configuration files are available in the wiki pages of the publicly available github sites (see below for links). We divide the description of the tool into three parts: (1) *the container,* (2) *the infrastructure* required to run the procedures in a computationally reproducible system, the neuroinformatics platform[30]; and, (3) *the DWI data analysis pipeline.* The main point is that whichever neuroinformatics platform is used, the results will be the same because the computations inside the containers will be the same.

MRI data acquisition. A total of 112 healthy volunteers (mean age = 24.4 years, SD = 4.22 years; 64 females) participated in the study. Twenty-four of the volunteers (mean age = 24.7 years, SD = 4.06 years; 13 females) returned for a second session in which they were scanned using the exact same MRI protocol (mean interval = 15 days, SD = 21.82 days, range: 7–104 days). All participants were right-handed and had normal or corrected-to-normal vision. No participant had a history of major medical, neurological disorders, or treatment for psychiatric disorders. The study protocol was approved by the Ethics Committee of the Basque Center for Cognition, Brain and Language (BCBL) and was carried out in accordance with the Code of Ethics of the World Medical Association (Declaration of Helsinki) for experiments involving human participants. Prior to their inclusion in the study, all subjects provided informed written consent. Participants received monetary compensation for their participation.

Whole-brain MRI data acquisition was conducted on a 3-T Siemens Prisma Fit whole-body MRI scanner (Siemens Medical Solutions) using a 64-channel whole-head coil. The MRI acquisition included one T1-weighted structural image (T1w) and two DWI sequences. High-resolution MPRAGE T1-weighted structural images were collected with the following parameters: time-to-repetition (TR) = 2530 ms, time-to-echo (TE) = 2.36 ms; flip angle (FA) = 7°, field of view (FoV) = 256 mm, voxel resolution = 1 mm^3, 176 slices. Both DWIs shared the following parameters: TR = 3600 ms, TE = 73 ms, FA = 78°, voxel resolution = 2 mm^3, 72 slices with no gap and a multiband acceleration factor of 3. The first DWI acquisition of 105 volumes and anterior to posterior phase-encoding consisted of 5 blocks of 21 volumes: one without diffusion weighting (b0; b-value of 0 s/mm^2), 10 with a b-value of 1000 s/mm^2 and 10 with a b-value of 2000s/mm^2, resulting in 50 isotropically distributed diffusion-encoding gradient directions for both the b = 1000 and b = 2000 shells. The five b0 images were acquired for motion correction. The second DWI acquisition of only seven b0 volumes was collected with reversed (posterior to anterior) phase-encoding to be used in geometrical distortion correction.

Container creation and configuration. The container creation process starts in a client machine (e.g., laptop, personal computer or development server). After testing that our analysis tool works locally (anatROIs, RTP2-preproc and RTP2-pipeline in our case), we start the Docker Image creation process. Docker is the platform where the Docker images run, and a running Image is called a Container, i.e. a Docker Container is the run-time instance of a Docker Image. The first step is selecting a base Image that includes the required features. For example, as RTP2-pipeline is a Matlab based program, we select a base Docker Image with the Matlab r2020b runtime, which is itself based on Ubuntu. In this base Docker Image, we install all the required programs, such as mrTrix and FSL (for an updated complete list of installed programs, refer to the Dockerfile file in https://github.com/garikoitz/RTP-pipeline). At this point, if it is a Matlab based program we compile it, otherwise, we build the Docker Image. Building the Image means obtaining all the required files and programs, including the compiled Matlab program, if required.

The full cycle of testing a container has the following steps (in this example we show RTP2-pipeline, which uses Matlab; anatROIs and RTP2-preproc are simpler as they do not use Matlab): (1) Run the code directly in Matlab and check the outputs; (2) Compile the Matlab code and run it using Matlab's Runtime environment (it is freely downloadable from www.mathworks.com); (3) Build the Docker Container and run it using the default parameters and changed parameters; (4) Check the code in github, tag it, and build the containers in dockerhub; (5) Download the containers in another machine and check with test data that they run appropriately. After this step, we can tag the container and upload it to Dockerhub, so that it can be shared with other users.

Infrastructure for computational reproducibility. RTP2 is organized in 3 containers that can be run in a number of environments. They can be run natively on any computer if the required software is installed as well. Below, we illustrate three possible ways of running the pipeline (see Fig. 1, bottom panels) using example cases.

Single computer. Examples of this computing platform are: (1) a single computer in the scanning room that is used for diffusion tractography, (2) a personal computer/laptop, (3) a small server in the lab/office used for

computing. In this case, we recommend having a Docker client implementation. Docker can be freely installed from https://www.docker.com/ by any user with administrator privileges. We tested it in macOS and Linux, and it should work in Windows too. The installation is typically simple, and it is done through the Docker client, searching for the containers nipy/heudiconv, garikoitz/anatrois, garikoitz/rtppreproc, garikoitz/rtp-pipeline, or using the command line:

```
docker pull nipy/heudiconv
docker pull garikoitz/anatrois
docker pull garikoitz/rtppreproc
docker pull garikoitz/rtp-pipeline
```

We recommend checking the latest version before installing. Once the containers are installed they are ready to run. For specific instructions about how to launch the containers, please see the wiki.

For individual subjects, it is manageable to launch the containers manually in the command line. This would be the recommendation for the testing phase. In the operational phase, we recommend a data organization and container launching scheme programmed in python. It can be obtained here https://github.com/garikoitz/launchcontainers. The instructions on how to use it are in the wiki of the repository. In order to save space, it creates symbolic links to files and takes care of the versioning of the files. Once the containers end the computations, the same computer can be used for tract visualization, quality assurance, or statistical analyses.

High performance computing (HPC) systems. It is common for many research labs to have access to HPC systems. These systems will be maintained by IT professionals, and typically will provide the researchers/clinicians with a way of accessing the system and running their data. To the best of our knowledge, such systems do not allow running Docker containers, but they accept running Singularity containers. The procedure is almost the same as for the Docker containers. Once the DICOMs files from the scanner have been transferred to the HPC system, it is possible to run results there and copy the results back to computers for visualization and statistical analysis.

Installation is as simple as with the Docker containers. Importantly, it does not require assistance from IT professionals. Below are the commands to install the containers in the home folder of the researcher:

```
singularity build $HOME/heudiconv.sif docker://nipy/heudiconv
singularity build $HOME/anatrois.sif docker://garikoitz/anatrois
singularity build $HOME/rtppreproc.sif docker://garikoitz/rtppreproc
singularity build $HOME/rtp-pipeline.sif docker://garikoitz/rtp-pipeline
```

Once installed, we recommend using the same python code in https://github.com/garikoitz/launchcontainers to prepare the data and launch the containers, although it is possible to launch everything manually. Please refer to the wiki for more instructions.

Cloud based neuroinformatics platform (Flywheel.io). Cloud based systems and software-as-a-service (SaaS) systems are taking over the software service landscape. The analysis containers were developed to be used as part of the neuroinformatics platform called Flywheel (and are extensively used this way). If your institution provides a subscription, you will be able to install these containers and use them as part of an integrated cloud-based tool (anybody with access to Flywheel can install RTP2 through the Flywheel Exchange: https://flywheel.io/gear-exchange/). Flywheel.io implements reproducible computational methods, tracks provenance of the data, and facilitates data sharing. The main functionalities provided by Flywheel are: *(1) Data Capture:* automatic data conversion and uploading from the scanner*, (2) Data Curation:* visualization, computation, sharing, download, reorganization, sharing, and renaming, can be done with remote tools (for example https://github.com/vistalab/scitran/), (3) *Computing:* running containers; every analysis, including versions, parameters, inputs, and outputs, is named, time-stamped and stored in the searchable database. (4) *Collaborating:* all the data, metadata and analyses can be made available to anyone granted permission in the organization. Containers are called gears in Flywheel. The same containers created in the first section are used as gears. For a detailed Gear creation and installation process, please refer to Flywheel's official documentation in https://docs.flywheel.io. See Figure S4 for details in Flywheel's technical architecture.

An important component of the whole RTP2 solution is that it allows programmatic access and control of the whole system. The same thing is true when RTP2 is run in the Flywheel system. We provide Matlab and Python based software similar to the *launchcontainers* software to control the cloud engine. In Flywheel the data storage and the computation jobs are always done in the cloud server. For a few subjects the interaction with Flywheel can be done through the web interface and most of the settings and tools can be tested this way. But increasingly, neuroimaging projects consist of a larger number of subjects, and programmatic access is required for time efficient and error free data, configuration parameters and result management. Even with two subjects, it is easy to make mistakes when setting the more than 50 RTP2 config parameters. To solve this problem we developed the Scitran tool. Scitran is a Matlab application built on top of Flywheel's Matlab SDK, that provides

several functionalities: *(1) Data/analysis results search:* to perform any of the following operations, *(2) Metadata update:* for example, socioeconomic information for the subjects, *(3) Upload and download information:* log files, images, or other project level required information*, (4) Job launch and management:* select a gear, version, config parameters and launch an analysis job and control the status, *(5) Information extraction:* select the desired subjects and analysis and extract only the required results for statistical analysis. Specifically for RTP2, there is a utility that extracts the per-subject-per-tract metric (e.g. FA) and creates a table with all the project level, acquisition level and analysis level parameters. Scitran is publicly available and can be installed from (https://github.com/vistalab/scitran/). Installing it in Matlab is as simple as downloading the zip file or cloning the repository and adding it to the Matlab path. Scitran comes with the latest Flywheel Matlab SDK. In order to make it work, it is required to have an active user in a Flywheel instance and download an API key. Please refer to the Scitran manual (https://github.com/vistalab/scitran/wiki) for further information on installing and using the Scitran tool.

Data analysis pipeline.   *anatROIs.*   The first step of the RTP2-pipeline (called RTP2-anatROIs) involves the processing of the subject's anatomical T1w image, with the objective of obtaining the ROIs that will later be used in tractography. The input of this step is the subject's T1w file and ROIs defined in MNI space; the output is a segmented T1w image and the ROIs of interest in individual subject T1w space. AnatROIs provides many predefined ROIs, and it is possible to extend it in multiple ways (see Fig. 2; https://github.com/garikoitz/anatROIs/wiki).

Included ROIs the T1w image is processed with the Freesurfer package (http://surfer.nmr.mgh.harvard.edu/) to obtain a cortical/subcortical segmentation and parcellation. If a previous freesurfer run exists, it is possible to use that, just passing the folder or a zip file with the completed analysis. This step already provides all of Freesurfer's segmentations and parcellations (check freesurfer or the wiki for more detail). Importantly, the surface parcellations are converted automatically to volumetric ROIs so that they can be used in tracking automatically. To convert ROIs or atlases defined in MNI space to individual-subject space, the container performed a non-linear registration of a 1 mm^3 MNI template using Advanced Normalization Tools (ANTs, http://stnava.github.io/ANTs/). In addition to all the Freesurfer segmentation (aseg, hippocampus, thalamus, brainstem) and parcellation (aparc2009) tools, we provided several ROIs in MNI space, the human connectome project (HCP) atlas[31] and cerebellum parcellation proposed by[32]. To parcellate the visual cortex we run the Neuropythy[33] tool on top of Freesurfer's results. Check the wiki for the latest version's available ROIs and segmentation tools (https://github.com/garikoitz/anatROIs/wiki/How-to-use).

User generated ROIs there are many ways to add ROIs or atlases to anatROIs. *(1) In template space:* MNI ROIs can be directly added to the container, as well as surface ROIs drawn in Freesurfer's fsaverage. *(2) Native space:* similarly, ROIs can be drawn in native space, either in 3D space or in the surface. The ROIs can come from analyses in other modalities, for example fMRI results. Please refer to the wiki for more details (https://github.com/garikoitz/anatROIs/wiki).

Computational load it varies greatly depending on the ROI selection and whether Freesurfer's recon-all has to be run. Running recon-all can take between 10 and 20 h depending on the subjects, resolution, and systems used. Then each set of ROIs has a different time requirement (ANTs registration takes a lot of time, for example). When selecting all ROIs, the researcher should consider allocating 24 h of computing time per subject as a precaution.

RTP2-preproc.   The preprocessing container prepares the data for the posterior DWI analysis in the RTP2-pipeline. The preprocessing does not include data analysis: its objective is to correct known problems in the DWI acquisitions process. It can optionally align the DWI data with the anatomical file (done by default). This gear is an adaptation from mrTrix's (https://github.com/MRtrix3/mrtrix3) recommended preprocessing procedure which was originally containerized in Brainlife (https://brainlife.io/app/5a813e52dc4031003b8b36f9). This container takes some required and optional inputs (listed below). More details in the wiki of the repository: https://github.com/garikoitz/RTP-preproc/wiki.

- Required:

  - Diffusion data in 4D Nifti format
  - Bvec file in FSL format
  - Bval file in FSL format

- Optional:

  - Anatomical T1w file: used to optionally align the diffusion data to it
  - Reversed phase encoding DWI data.
  - Bval file, corresponding to the reversed phase encoding DWI data.
  - Bvec file, corresponding to the reversed phase encoding DWI data.
  - Brainmask: it can be Freesurfer's brainmask or any mask edited manually. If a mask is provided, the container does not use FSL's BET tool, which usually doesn't give very good results.

It is highly recommended to acquire the reversed phase encoding files to correct for EPI geometric distortions with FSL's TOPUP. If those images are not provided, the alignment between the diffusion data and the anatomy will not be as good. This can be relevant downstream for analysis requiring a good alignment between
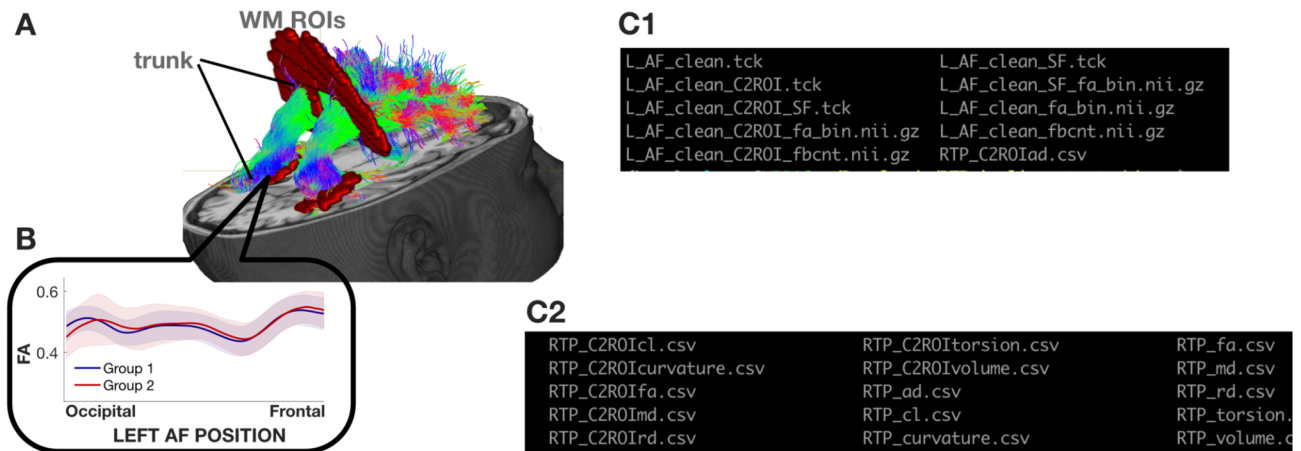
Figure 5. Illustrative tract (arcuate fasciculus) with defining ROIs and tract profile. (**A**) The streamlines serve as a model of white matter tracts; they are selected by fitting to the diffusion-weighted imaging (DWI) measurements. The tracts are defined by regions of interest (ROIs, red) that select specific streamlines from the whole brain tractogram (in this example). The region between the two ROIs is relatively stable and called the trunk. (**B**) Tract's FA profile. We can calculate profiles from the whole tract or only the trunk, called clip-to-roi (C2ROI) operation. We estimate a core fiber (called super fiber—SF) from the collection of streamlines and sample 100 equally spaced segments. The FA of the core fiber is calculated by combining FA transverse to the core fiber at every sample point, using a Gaussian weighting scheme over distance. The set of sample points is the tract profile; the average of the FA values of the core fiber is the mean tract FA. In the inset, an illustrative example of two group profiles. The dark outline is the mean of the groups, and the shaded outline represents one standard deviation around these values. The plots are generated with Matlab scripts created by reading the csv file values described in (**C2**). (**C1**) List of files in the output of the RTP2-pipeline. Every tract generates a number of different outputs, for both the C2ROI and normal options. Tck files are tract definitions, and they can be the full tract or only the super-fiber (marked as _SF). On top of the tck-s, we created two types of Nifties, the fa_bin Nifti is a mask of the tract, with 1-s where there are fibers and 0-s elsewhere. In the fbcnt Nifti it is zero where there are no fibers, and in the rest of the voxels the value is the fiber count in the voxel. (**C2**) The csv files contain the tract profiles for all of the different tracts that were tracked in this specific subject. There is one file per metric. Thus, the size of every csv is $M^{row} \times N^{column}$ (M: 100, N: number of tracts obtained in this analysis).

anatomical and diffusion data—for example if we want to use mrTrix's Anatomically Constrained Tractography, or any cortical ROI.

In the wiki we provide an exhaustive list of the preprocessing steps that can be set using the configuration parameters (https://github.com/garikoitz/RTP-preproc/wiki/configjson). We will use an application logic order here; this is the chronological order of how the options are applied. Once the analysis is run, it is recommended to visually inspect the results, if not in all the subjects, in a representative number of them, to avoid any systematic and database dependent errors. Diffusion MRI suffers from a range of imaging artifacts. This research field is really active, providing improved solutions for correcting them[34]. Our system implements the latest steps in data preprocessing, and with a configuration file it is possible to select which one of the steps is being executed. New solutions are integrated in the gear in new versions, and the containerized implementation allows for the systematic testing of all the new options.

Computational load it depends on the voxel size, the hardware and the selected options. The researcher should consider allocating 12 h of computing time and then adjusting according to the dataset and the options selected.

RTP2-pipeline. The tractography container takes preprocessed DWI data from RTP2-preproc and the ROIs from anatROIs and automatically identifies any number of user-specified white matter tracts. Using the ROIs and atlases provided by default (check the wiki for an updated list https://github.com/garikoitz/anatROIs/wiki/atlases) or by adding new user defined ROIs, the tracking possibilities are endless. Furthermore, it is possible to do some ROI post-processing operations within the tool. Basically, we can perform AND operations between ROIs to generate bigger ROIs from existing ones, and we can dilate ROIs to make them bigger (e.g., with gray-matter ROIs it is desirable to dilate them to be sure that the white matter fibers reach the ROIs and can be detectable). Once the tracts are identified, it provides the tract and its metrics in several different ways (see Fig. 5). In the analysis process, the pipeline generates values at the voxel and the tract level, which can later be used for further analysis.

There are two main approaches to generate one tract: edit from whole brain streamlines or generate originally from given ROIs. The recommendation is to apply the first approach, when possible, because whole-brain streamlines can be validated and trimmed by specific algorithms (see details described next). In the current version (RTP2), the RTP2-pipeline is heavily based on mrTrix. The list of parameters is too long to show here, for a complete list please refer to the manifest.json file in (https://github.com/garikoitz/RTP-pipeline/wiki). The wiki contains an example_config.json file with all the default values (for Flywheel, the manifest.json contains a description and a default value for every parameter).

It is important to notice that the parameter selection is project specific. We recommend using the default in a first test run, but from that moment on, it is usually required to adjust the parameters. The github repository contains a wiki page with parameter recommendations (https://github.com/garikoitz/RTP-pipeline/wiki/Parameter-recommendations) for different types of acquisition data, which researchers should find useful. We hope that this list of recommendations will grow in the future.

This container has a list of required files:

- Preprocessed diffusion data (output from RTP2-preproc, ideally)
- Anatomical T1w image (output from anatROIs, ideally)
- Zip file containing the ROIs (output from anatROIs, ideally, but it can be created manually, or use any existing atlas in MNI or fsaverage to create the subject space ROIs).
- Config.json file: it will be used mostly for the whole-brain tractography.
- Tractparams.csv file: it indicates all of the tracts to be obtained, detailing the ROIs and the configuration parameters. See https://github.com/garikoitz/RTP-pipeline/wiki/How-to-edit-tractparams.csv for an example and instructions on how to complete it. If a new acquisitions sequence or special population is being used, it will require adjustments the first time it is run. The ROI names used here refer to the ROI names in the zip file.

Next, we illustrate the main steps implemented in the pipeline and explain the main configuration parameter choices:

1. ROI validation. Check existence of ROI files, and dilate/combine ROIs if specified in the tractparams.csv file (see https://github.com/garikoitz/RTP-pipeline/wiki/How-to-edit-tractparams.csv).
2. DTI modeling at the voxel level and metric creation: FA, MD, RD etc.
3. Response function creation. Different algorithms can be selected, and with different options. We recommend using the dhollander algorithm with automatic lmax, which separates the WM, GM, and CSF in the DWI without requiring an anatomical file.
4. Constrained Spherical Deconvolution (CSD) modeling. Different algorithms can be selected. Performed at the voxel level, it creates Fiber Orientation Distributions (FOD) that are required for tracking.
5. Whole-brain white-matter streamlines estimation. Whole-brain streamlines will be estimated when there is at least one tract being generated from editing whole-brain streamlines (as specified in tractparams.csv file). Different algorithms and options are available (see mrTrix documentation for a comprehensive list on the tckgen command). Additionally, there are other decisions that can be taken at this point. We implemented Ensemble Tractography[35], SIFT[36] and LiFE (Linear Fascicle Evaluation)[37] into the RTP2-pipeline. It is possible to run them independently (i.e. both, one or the other, or none can be selected).

   a. Ensemble Tractography (ET) method: ET invokes MRtrix's tractography tool a number of times (selected with a configuration parameter), constructing whole-brain tractograms with a range of tracking parameter options. For example, the minimum angle parameters for tracking can be varied (e.g. 47, 23, 11, 6, 3), and the resulting whole-brain connectomes concatenated together. There are config parameters specific to the Ensemble Tractography implementation.
   b. The SIFT and LiFE (Linear Fascicle Evaluation) methods evaluate the tractogram streamlines and retain those that meaningfully contribute to predicting variance in the DWI data. There are config parameters specific to the SIFT and LiFE implementation.

6. Tracts generation. User-defined tracts can be defined either from selecting specific streamlines from the whole-brain tractogram (see previous step), or generating streamlines using tckgen in MRtrix using seed and target ROIs.
7. Tract Profiles. The voxel-based metrics are calculated for each individual tract creating the tract profiles. The metrics can be of any voxel-based measurement. Although the most common ones are the DTI metrics (FA, MD, …), we have successfully used others, such as T1 relaxation time or macromolecular tissue volume (MTV) quantitative MRI maps. The basic steps for tract profile creation are:

   a. A core fiber, representing the central tendency of all the streamlines in the tract, is identified.
   b. Equally spaced positions along the fiber between the two defining ROIs are sampled (N = 100).
   c. The metric values of streamlines at locations transverse to each sample position are measured and combined. The combination is weighted by a Gaussian value based on the distance from the sample point.
   d. The sampling and transverse averaging generates a tract profile of 100 metric values (see inset in Fig. 5 for a group mean FA tract profile).
   e. The tract profiles are outputted in csv format. There is one file per metric and subject. The size of every csv is Mrow x Ncolumn (M: 100, N: number of tracts obtained in the analysis). The csv files can readily be incorporated in any statistical analysis package.

8. Tract cleaning. After generating the tract, and calculating the core fiber, it is possible to clean the tract by removing any streamlines for which (1) the length is a specific number of standard deviations away from the mean fiber length or (2) the distance to the core fiber is beyond a specific number of standard deviations from the mean distance of all streamlines. This means that fiber groups will be forced to be a compact bundle.

9. Export tracts as Nifti masks (see Fig. 5-C1, marked as '_bin' files). It creates volumetric images in native and template space, creating binary masks (1 if a fiber crosses the voxel, 0 if not), or fiber count masks (each voxel indicates the amount of fibers crossing the voxel).

Computational load in addition to all the considerations from previous steps (voxel size, hardware, analysis options…), this step greatly depends on the number of tracts that are to be obtained. To obtain one roi-to-roi tract, it should be sufficient to allocate 30 min of computing time. In order to obtain several WBT tracts and roi-to-roi tracts, it may be prudent to allocate 48 h of computing time and then adjust if necessary.

Data availability

All data used in the present work will be shortly uploaded to an open data repository (https://osf.io/sgmuj/). All the data used in the present work was collected at our research center (BCBL) using the same 3-T Siemens Prisma Fit MRI scanner and MRI protocols. Prior to their inclusion in the study, all participants provided informed written consent. Data collection was approved by the Ethics Committee of our center (BCBL) and was carried out in accordance with the Code of Ethics of the World Medical Association (Declaration of Helsinki) for experiments involving human participants.

References

1. Wilson, G. *et al.* Good enough practices in scientific computing. *PLoS Comput. Biol.* **13**, e1005510 (2017).
2. Sandve, G. K., Nekrutenko, A., Taylor, J. & Hovig, E. T. simple rules for reproducible computational research. *PLoS Comput. Biol.* **9**, e1003285 (2013).
3. Buckheit, J. B. & Donoho, D. L. WaveLab and Reproducible Research. In *Wavelets and Statistics* (eds. Antoniadis, A. & Oppenheim, G.) 55–81 (Springer New York, 1995).
4. Stodden, V., Leisch, F. & Peng, R. D. *Implementing Reproducible Research* (Chapman and Hall/CRC, 2018).
5. Peng, R. D. Reproducible research in computational science. *Science* **334**, 1226–1227 (2011).
6. Botvinik-Nezer, R. *et al.* Variability in the analysis of a single neuroimaging dataset by many teams (Preprint). (2020).
7. Schilling, K. G. *et al.* Tractography dissection variability: What happens when 42 groups dissect 14 white matter bundles on the same dataset?. *Neuroimage* **243**, 118502 (2021).
8. Yarkoni, T. & Westfall, J. Choosing Prediction over explanation in psychology: Lessons from machine learning. *Perspect. Psychol. Sci.* **12**, 1100–1122 (2017).
9. Baker, M. 1500 scientists lift the lid on reproducibility. *Nature* **533**, 452–454 (2016).
10. Tournier, J.-D. *et al.* MRtrix3: A fast, flexible and open software framework for medical image processing and visualisation. *Neuroimage* **202**, 116137 (2019).
11. Smith, S. M. *et al.* Advances in functional and structural MR image analysis and implementation as FSL. *Neuroimage* **23**, S208–S219 (2004).
12. Yendiki, A. *et al.* Automated probabilistic reconstruction of white-matter pathways in health and disease using an atlas of the underlying anatomy. *Front. Neuroinform.* **5**, 23 (2011).
13. Wang, R., Benner, T., Sorensen, A. G. & Wedeen, V. J. Diffusion toolkit: A software package for diffusion imaging data processing and tractography. in *Proceedings of the International Society for Magnetic Resonance in Medicine* Vol. 15 (Berlin, 2007).
14. Yeh, F.-C., Wedeen, V. J. & Tseng, W.-Y.I. Generalized q-Sampling Imaging. *IEEE Trans. Med. Imaging* **29**, 1626–1635 (2010).
15. Garyfallidis, E. *et al.* Dipy, a library for the analysis of diffusion MRI data. *Front. Neuroinform.* **8**, 8 (2014).
16. Theaud, G. *et al.* TractoFlow: A robust, efficient and reproducible diffusion MRI pipeline leveraging Nextflow & Singularity. *Neuroimage* **218**, 116889 (2020).
17. Wasserthal, J., Neher, P. & Maier-Hein, K. H. TractSeg: Fast and accurate white matter tract segmentation. *Neuroimage* **183**, 239–253 (2018).
18. Yeatman, J. D., Dougherty, R. F., Myall, N. J., Wandell, B. A. & Feldman, H. M. Tract profiles of white matter properties: Automating fiber-tract quantification. *PLoS ONE* **7**, e49790 (2012).
19. Kruper, J. *et al.* Evaluating the reliability of human brain white matter tractometry. *Apert. Neuro* https://doi.org/10.52294/e6198 273-b8e3-4b63-babb-6e6b0da10669 (2021).
20. Lerma-Usabiaga, G., Perry, M. L. & Wandell, B. A. Reproducible Tract Profiles (RTP): From diffusion MRI acquisition to publication. *bioRxiv* https://doi.org/10.1101/680173 (2019).
21. Lerma-Usabiaga, G., Mukherjee, P., Perry, M. L. & Wandell, B. A. Data-science ready, multisite, human diffusion MRI white-matter-tract statistics. *Sci. Data* https://doi.org/10.1038/s41597-020-00760-3 (2020).
22. Liu, M., Lerma-Usabiaga, G., Clascá, F. & Paz-Alonso, P. M. Reproducible protocol to obtain and measure first-order relay human thalamic white-matter tracts. *Neuroimage* **262**, 119558 (2022).
23. Fischl, B. FreeSurfer. *Neuroimage* **62**, 774–781 (2012).
24. Gorgolewski, K. J. *et al.* The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments. *Sci. Data* **3**, 160044 (2016).
25. Pierpaoli, C., Jezzard, P., Basser, P. J., Barnett, A. & Di Chiro, G. Diffusion tensor MR imaging of the human brain. *Radiology* **201**, 637–648 (1996).
26. Van Essen, D. C. The human connectome project. *J. Vis.* **11**, 8–8. https://doi.org/10.1167/11.15.8 (2011).
27. Grisot, G., Haber, S. N. & Yendiki, A. Diffusion MRI and anatomic tracing in the same brain reveal common failure modes of tractography. *Neuroimage* **239**, 118300 (2021).
28. Bonilha, L. *et al.* Reproducibility of the structural brain connectome derived from diffusion tensor imaging. *PLoS ONE* **10**, e0135247 (2015).
29. Khalsa, S., Mayhew, S. D., Chechlacz, M., Bagary, M. & Bagshaw, A. P. The structural and functional connectivity of the posterior cingulate cortex: Comparison between deterministic and probabilistic tractography for the investigation of structure–function relationships. *Neuroimage* **102**, 118–127. https://doi.org/10.1016/j.neuroimage.2013.12.022 (2014).
30. Marcus, D. S. *et al.* Informatics and data mining tools and strategies for the human connectome project. *Front. Neuroinform.* **5**, 4 (2011).
31. Glasser, M. F. *et al.* The human connectome project's neuroimaging approach. *Nat. Neurosci.* **19**, 1175–1187 (2016).
32. Diedrichsen, J. *et al.* Imaging the deep cerebellar nuclei: A probabilistic atlas and normalization procedure. *Neuroimage* **54**, 1786–1794 (2011).

33. Benson, N. C. & Winawer, J. Bayesian analysis of retinotopic maps. *Elife* **7**, e40224 (2018).
34. Tax, C. M. W., Bastiani, M., Veraart, J., Garyfallidis, E. & Okan Irfanoglu, M. What's new and what's next in diffusion MRI pre-processing. *Neuroimage* **249**, 118830 (2022).
35. Takemura, H., Caiafa, C. F., Wandell, B. A. & Pestilli, F. Ensemble tractography. *PLoS Comput. Biol.* **12**, 1–22 (2016).
36. Smith, R. E., Tournier, J. D., Calamante, F. & Connelly, A. S. I. F. T. Spherical-deconvolution informed filtering of tractograms. *Neuroimage* **67**, 298–312 (2013).
37. Pestilli, F., Yeatman, J. D., Rokem, A., Kay, K. N. & Wandell, B. A. Evaluation and statistical inference for human connectomes. *Nat. Methods* **11**, 1058–1063 (2014).

Competing interests

Brian Wandell works with Flywheel Exchange, LLC.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-023-32924-7.

Correspondence and requests for materials should be addressed to G.L.-U.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.