Confluence Documentation

Module Architecture

November 17, 2021

Nikki Tebaldi (ntebaldi@umass.edu)

TABLE OF CONTENTS

TABLE OF FIGURES

ECOSYSTEM

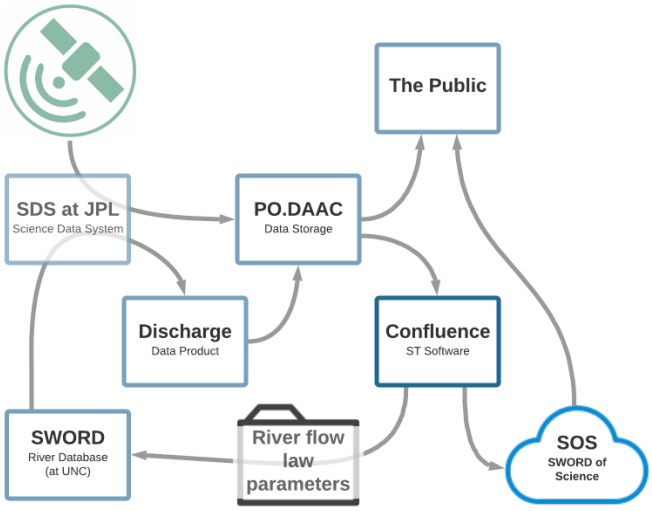

Figure 1 Confluence Ecosystem

Confluence exists within a larger NASA ecosystem. Surface Water and Ocean Topography (SWOT) observations are processed into shapefiles by the Science Data System (SDS) at the Jet Propulsion Laboratory (JPL) and stored in an S3 bucket by the Physical Oceanography Distributed Active Archive Center (PO.DAAC). Confluence transfers data from the S3 bucket, processes it, and outputs results to the SWORD of Science (SoS) which has been designed, developed, and hosted by UMass Amherst. Confluence also outputs flow law parameters to the SDS which uses the parameters to produce discharge estimates. These estimates are stored in the SWOT River Database (SWORD) which has been designed, developed, and hosted by UNC. The discharge estimates make up the SWOT mission data product and are stored in PO.DAAC and are used by the next execution of Confluence.

MODULES

OVERVIEW

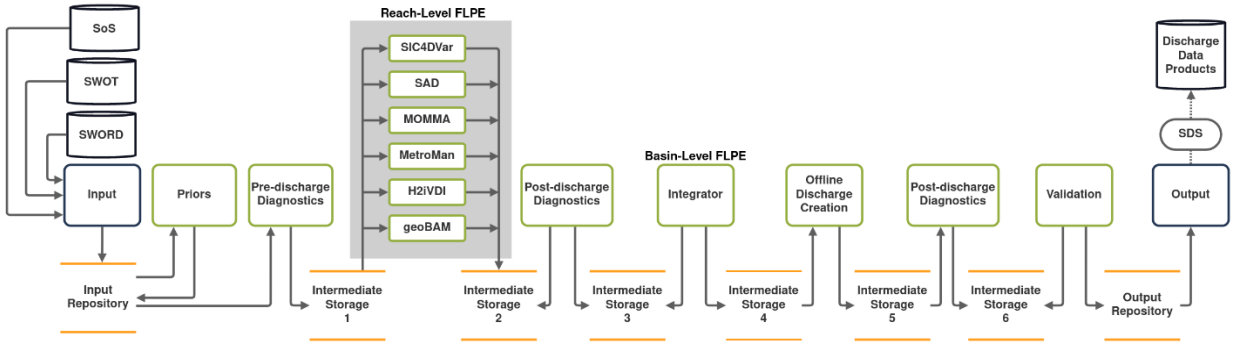

Figure 2 Confluence Data Flow Diagram

Confluence is made up of many components that process data along a data "pipeline". The pipeline starts with the input data, various modules process this data, and then the results of this processing are stored. Confluence starts with the input module which takes SWOT shapefiles and transforms them into a time series format expected by all the other modules. The pipeline then proceeds through the generation of priors data, pre diagnostics, reach-level flow law parameter estimation (FLPE), post diagnostics, integration, offline discharge creation, and validation. The pipeline ends with the output module which stores the results of each module in a new version of the SoS and uploads the new version to an Amazon Simple Storage Service (S3) bucket.

INPUT MODULE

Description	The Input module takes SWOT pass/cycle observation data from PO.DAAC's AWS infrastructure (S3 bucket) and transforms it into a matrix with nx by nt dimensions where nx is the number of spatial units and nt is the number of observations. The data is stored by reach in NetCDF files.	
Input	S3 URI, PO.DAAC EarthData credentials, S3 credentials, SWOT shapefiles	
Output	NetCDF file (one per reach) with the following reach and node data:	
	Reach slope2, slope2_u, width, width_u, wse, wse_u, d_x_area, d_x_area_u, reach_q, dark_frac, ice_clim_f, ice_dyn_f, partial_f, n_good_nod, obs_frac_n, xovr_cal_q	**Node** slope2, slope2_u, width, width_u, wse, wse_u, d_x_area, d_x_area_u, noder_q, dark_frac, ice_clim_f, ice_dyn_f, partial_f, n_good_nod, obs_frac_n, xovr_cal_q
Processing-level	Under development (parallelization could occur on continent or granule level)	
Command line arguments	n/a	
Code Repository	https://github.com/SWOT-Confluence/input	

PRIORS MODULE

Description	Generates new prior data (GRDC, USGS, geoBAM Priors) and overwrites GRADES data with gaged data for 'constrained' product executions.
Input	SoS prior NetCDF files
Output	SoS NetCDF files with updated priors (new version uploaded to S3 bucket)
Processing-level	continent
Command line arguments	1 run_type (required): 'unconstrained' or 'constrained' 2 grdc (optional): 'grdc' 3 usgs (optional): 'usgs' 4 gbpriors (optional): 'gbpriors'
Code Repository	https://github.com/SWOT-Confluence/priors/blob/main/update_priors.py

PREDIAGNOSTICS MODULE

Description	Parses the SWOT observation data formatted by the Input module to track quality indicators and perform consistency checks. Replaces flags with missing values.
Input	SWOT NetCDF files
Output	Overwrites SWOT NetCDF files and outputs flags in separate reach NetCDF files (only outputs data for reaches with valid data)
Processing-level	reach-level (SWOT NetCDF file)
Command line arguments	1 json_file (optional): 'reaches.json'
Code Repository	https://github.com/SWOT-Confluence/prediagnostics

GEOBAM (FLPE) MODULE

Description	Take the SWOT observation data, SoS priors, and SWORD data to produce discharge time series and discharge parameters.
Input	SWOT: d_x_area (node), slope2 (node), width (node) SoS: mean_q
Output	logq_m, logq_sd, logn_man_mean, logn_man_sd, logn_amhg_m, logn_amhg_sd, A0_m, A0_sd, logWc_m, logWc_sd, logQc_mean, logQc_sd, b_m, b_sd, logr_m, logr_sd, logWb_m, logWb_sd, logDb_m, logDb_sd
Processing-level	reach-level (SWOT NetCDF file)
Command line arguments	1 json_file (optional): 'reaches.json'
Code Repository	https://github.com/SWOT-Confluence/geobam https://github.com/SWOT-Confluence/geobamdata

H2iVDI (FLPE) Module

Description	Take the SWOT observation data, SoS priors, and SWORD data to produce discharge time series and discharge parameters.
Input	SWOT: wse, width, slope2 SoS: mean_q SWORD: ?
Output	A0, alpha, beta, Q
Processing-level	reach-level (SWOT NetCDF file)
Command line arguments	1 -reachjson (optional): '-reachjson reaches.json' 2 -chain (optional): '-chain classic'
Code Repository	https://github.com/SWOT-Confluence/h2ivdi

MetroMan (FLPE) Module

Description	Take the SWOT observation data, SoS priors, and SWORD data to produce discharge time series and discharge parameters.
Input	SWOT: wse (reach), width (reach), slope2 (reach) SoS: mean_q SWORD: reach_id, reach_length, dist_out
Output	A0, na, x1, allq, qu
Processing-level	inversion sets (sets of SWOT NetCDF files)
Command line arguments	1 json_file (optional): 'reaches.json'
Code Repository	https://github.com/SWOT-Confluence/metroman

MOMMA (FLPE) Module

Description	Take the SWOT observation data, SoS priors, and SWORD data to produce discharge time series and discharge parameters.
Input	SWOT: width (reach), wse (reach), slope2 (reach) SoS: logDb_hat
Output	stage, width, slope, Qgage, seg, n , Y, v, Q, Q_constrained, gage_constrained, input_MBL_prior, input_QM_prior, input_QB_prior, input_Yb_prior, input_known_ezf, input_known_bkfl_stage, input_known_nb_seg1, input_known_x_seg1, Qgage_constrained_nb_seg1, Qgage_constrained_x_seg1, input_known_nb_seg2, input_known_x_seg2, Qgage_constrained_nb_seg2, Qgage_constrained_x_seg2, n_bkfl_Qb_prior, n_bkfl_final_used, vel_bkfl_Qb_prior, vel_bkfl_diag_MBL, Froude_bkfl_diag_Smean, width_bkfl_empirical, width_bkfl_solved_obs, depth_bkfl_solved_obs, depth_bkfl_diag_MBL, depth_bkfl_diag_Wb_Smean, zero_flow_stage, bankfull_stage, Qmean_prior, Qmean_momma, Qmean_momma.constrained
Processing-level	reach-level (SWOT NetCDF file)
Command line arguments	1 json_file (optional): 'reaches.json'
Code Repository	https://github.com/SWOT-Confluence/momma

SAD (FLPE) MODULE

Description	Take the SWOT observation data, SoS priors, and SWORD data to produce discharge time series and discharge parameters.
Input	SWOT: slope2 (node), wse (node), width (node) SoS: mean_q, min_q, max_q, logQ_sd, logn_hat, logn_sd, upperbound_logn, lowerbound_logn, logr_hat, logr_sd, upperbound_logr, lowerbound_logr, logDB_hat, logDB_sd, upperbound_logDB, lowerbound_logDb
Output	A0, n, Qa, Q_u
Processing-level	reach-level (SWOT NetCDF file)
Command line arguments	1 json_file (optional): 'reaches.json'
Code Repository	https://github.com/SWOT-Confluence/sad

SIC4DVAR (FLPE) MODULE

Description	Take the SWOT observation data, SoS priors, and SWORD data to produce discharge time series and discharge parameters.
Input	SWOT: width (node), d_x_area (node), slope2 (node), wse (node), d_x_area (reach), slope2 (reach), width (reach), wse (reach) SoS: mean_q
Output	A0, n, q_algo5, q_algo31, half_width, elevation
Processing-level	reach-level (SWOT NetCDF file)
Command line arguments	1 json_file (optional): 'reaches.json'
Code Repository	not available

MEAN OPTIMIZATION INTEGRATOR (MOI) (FLPE) MODULE

Description	Takes the formatted SWOT observation data and reach-level FLPE output and integrates the results. It uses river network topology to force mass conservation and also defines uncertainty.
Input	SWOT: wse (reach), width (reach), slope2 (reach), d_x_area (reach) FLPE output (parameters and discharge time series)
Output	geobam/q, geobam/a0, geobam/n, geobam/qbar_reachScale, geobam/qbar_basinScale hivdi/q, hivdi/a0, hivdi/n, hivdi/qbar_reachScale, hivdi /qbar_basinScale, metroman/q, metroman/a0, metroman/n, metroman/qbar_reachScale, metroman/qbar_basinScale, momma/q, momma/a0, momma/n, momma/qbar_reachScale, momma/qbar_basinScale, sad/q, sad/a0, sad/n, sad/qbar_reachScale, sad/qbar_basinScale, sic4dvar/q, sic4dvar/a0, sic4dvar/n, sic4dvar/qbar_reachScale, sic4dvar/qbar_basinScale,

Processing-level	basin
Command line arguments	1 json_file (optional): 'basin.json'
Code Repository	https://github.com/SWOT-Confluence/MOI

POSTDIAGNOSTICS MODULE

Description	**FLPE:** Compares previous and current discharge values while also performing a realism check on the results of each FLPE algorithms' execution. **MOI:** Compares FLPE and integrated discharge values while also performing a realism check on the results of Integrator execution. Also compares previous and current Integrator discharge.
Input	FLPE output: current discharge time series MOI output: current discharge time series SoS: mean_q, max_q, min_q, logQ_sd, previous FLPE discharge time series, previous MOI discharge time series
Output	FLPE: num_algos, algo_names, realism_flags, stability_flags
Processing-level	reach
Command line arguments	1 reaches_json (optional): 'reaches.json' 2 tolerance (optional): 0.25
Code Repository	https://github.com/SWOT-Confluence/postdiagnostics

OFFLINE MODULE

Description	Executes the RiverObs SWOT Algorithm that will be run by the Science Data System (SDS) to produce discharge for each FLPE algorithm and executes a consensus algorithm.
Input	SWOT: wse (reach), wse_u (reach), width (reach), width_u (reach), slope2 (reach), slope2_u (reach), d_x_area (reach), d_x_area_u (reach) FLPE results
Output	d_x_area, d_x_area_u, metro_q_c, bam_q_c, hivdi_q_c, momma_q_c, sads_q_c, consensus_q_c, metro_q_uc, bam_q_uc, hivdi_q_uc, momma_q_uc, sads_q_uc, consensus_q_uc
Processing-level	reach
Command line arguments	1 run_type (optional): 'unconstrained' or 'constrained' 2 reaches_json (optional): 'reaches.json'
Code Repository	https://github.com/SWOT-Confluence/offline-discharge-data-product-creation

VALIDATION MODULE

Description	Compares the results of FLPE and integrator modules to gage data, calculates statistics and creates hydrographs.
Input	SoS: reach_ids, usgs/q, usgs/q_t, grdc/q, grdc/q_t Offline results

Output	algorithm, NSE, Rsq, KGE, RMSE, testn jpg
Processing-level	continent
Command line arguments	1 cont_json (optional): continent.json'
Code Repository	https://github.com/SWOT-Confluence/Validation

OUTPUT MODULE

Description	Creates a new 'results' version of the SoS and uploads it to the 'confluence-sos' S3 bucket. Results from each module are stored in the SoS.
Input	NetCDF files from each module
Output	SoS_results.nc file for each continent
Processing-level	continent
Command line arguments	1 cont_json (optional): continent.json' 2 run_type (optional): 'constrained' or 'unconstrained'
Code Repository	https://github.com/SWOT-Confluence/output

List Of Acronyms

AWS: Amazon Web Services

EC2: Amazon Elastic Compute Cloud

ECR: Amazon Elastic Container Registry

EFS: Amazon Elastic File System

FLPE: Flow Law Parameter Estimation

IAM: AWS Identity and Access Management (IAM)

JPL: Jet Propulsion Laboratory

PO.DAAC: Physical Oceanography Distributed Active Archive Center

S3: Amazon Simple Storage Service

SDS: Science Data System

SoS: SWORD of Science

SWORD: SWOT River Database

SWOT: Surface Water and Ocean Topography

VPC: Amazon Virtual Private Cloud