Linking stormwater control performance to stream ecosystem outcomes: incorporating a performance metric into effective imperviousness

S4 Appendix. Calculation of effective imperviousness variants and water fluxes in study subcatchments

Christopher J. Walsh, Matthew J. Burns, Tim D. Fletcher, Darren G. Bos, Peter Poelsma, Joshphar Kunapo and Sam J. Imberger

School of Ecosystem and Forest Sciences, The University of Melbourne, 500 Yarra Boulevard, Burnley Victoria 3121, Australia

This document provides an overview of the code used to calculated effective imperviousness (EI) variants in experimental catchments, Little Stringybark Creek and Dobsons Creek. The code was written as a series of functions in R [1], drawing data from the database described in Appendix S3. The code for this document, and for all functions are available in the github repository https://github.com/cjbwalsh/lsc_dbs_scms, linked to the Open Science Framework repository [2]: https://osf.io/57azq/.

Calculation of EI in a catchment required separating impervious area draining directly to the stream from impervious areas draining to one or more stormwater control measures (SCMs). For the latter, terminal SCMs (those that drain directly to the stormwater system or to the stream) were identified, and S was calculated accounting for any upstream SCMs. To calculate a time series of EI in each catchment it was also necessary to account for changes in impervious coverage and the installation and operation of SCMs in each subcatchment. Over time, effective impervious coverage varied as buildings and paving were constructed, demolished or connected to the drainage network; SCMs were installed, decommissioned or their specifications were altered; relationships between SCMs changed as SCMs in treatment trains came on line; and in two cases, when large-scale SCMs diverted runoff from an upstream subcatchment (King Street Upper, and The Entrance raingardens) relationships between subcatchments changed. The upstream-downstream relationships between SCMs are illustrated in Fig. S4-1 for a point in time after all SCMs were installed, but such maps differed over the study period.

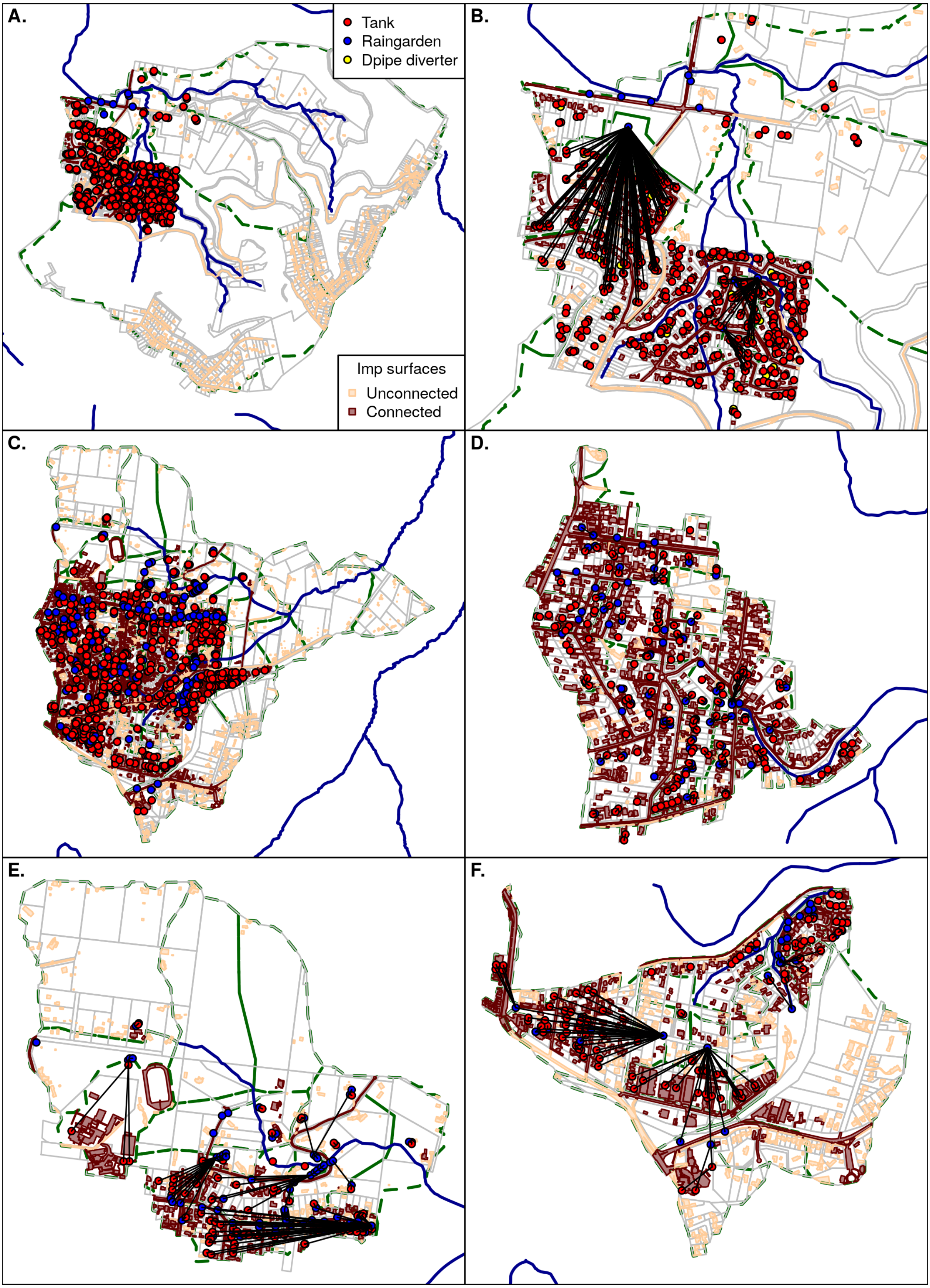

Fig. S4-1. Experimental catchments showing connected and unconnected impervious areas and stormwater control measures (SCMs), and in B, D-F, connections between SCMs. A. D8; B. Detail of that part of the D8 and D4 catchments serviced by stormwater drainage pipes draining to the stream; C. L4; D. L1; E. LN; F. LS.

Calculation of impervious runoff fluxes and EB in each subcatchment thus required adjustment of the database tables at each time of change to portray the distribution and relationships between impervious surfaces and SCMs. A group of related functions (Fig. S4-2) were thus developed to:

1. compile database tables for a given date,
2. identify the terminal SCMs in each subcatchment on that date (i.e. those SCMs that drained to the stormwater drainage network rather than to another SCM),
3. calculate EB of each terminal SCM on that date, or the water budget from that date until the next change, and the impervious area still draining directly to stormwater,
4. compile a time series of EB or water budget for each SCM in the subcatchment
5. calculate overall subcatchment statistics by summing results for all SCMs and the untreated impervious area.

Fig. S4-2. Primary functions in BACRIfunctions.R (and modelfunctions2017.R in the case of tankmodel and gardenmodel) and the flow of calculations to achieve the primary outputs of budget_subc_time_series and EI_subc_time_series.

Example calculations

The following uses Wattle Valley Rd North subcatchment (pipeID = 29) as an example for calculating subcatchment EB, EI and water fluxes.

It is a small catchment with a mix of connected and unconnected impervious surfaces (roofs and roads), with a treatment train of tanks and raingardens on each of two properties, with all impervious surfaces draining to a series of raingardens at the bottom of the catchment before entering Little Stringybark Creek North. This permits a nested series of examples, with and without the terminal series of raingardens (Fig. S4-3).

```
scmMap(example_pipeID)  #Mapping function in BACRIfunctions.R
box()
```

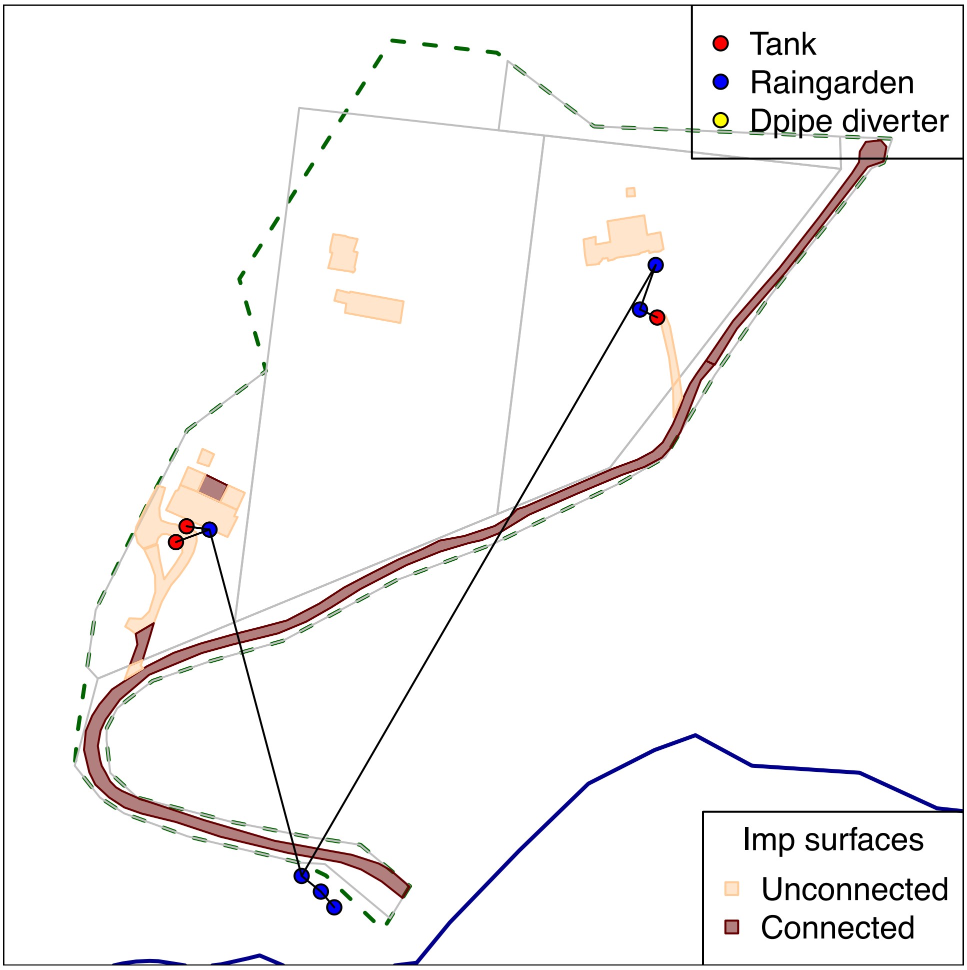

Fig. S4-3. Map of example subcatchment (Wattle Valley Rd North), with a black line indicating upstream-downstream relationships between SCMs. (Locations of SCMs are approximate within their correct land parcel.)

We begin by demonstrating the two primary functions for calculating time series of EI variants and water fluxes, before stepping through the sub-functions that they use in their calculations (Fig. S4-0).

Time series of EI variants and of modelled flows over the length of the study can simply be calculated and plotted using code shown below for Figs. S3 and S4. For this small catchment, the EI calculation (Fig. S4-3) took ~20 s, and the water flux calculation (Figs. S3) took ~90 s. Calculations for larger sub-catchments take between 2 (L1) and 50 min (L4).

```
Croydon <- prepare_runoff_data(get(load(paste(here::here("data"),
                        "/Croydon_1966_hourly_rain_runoff_et.rda", sep = ""))))
ei_29 <- EI_subc_time_series(29)
plotlEItrends(ei_29$iats, legend = TRUE, ylab = TRUE, xlab = TRUE, cex = 0.8)
title(ylab = "Effective imperviousness variant (%)", xlab = "Date", cex.lab = 0.75)
```

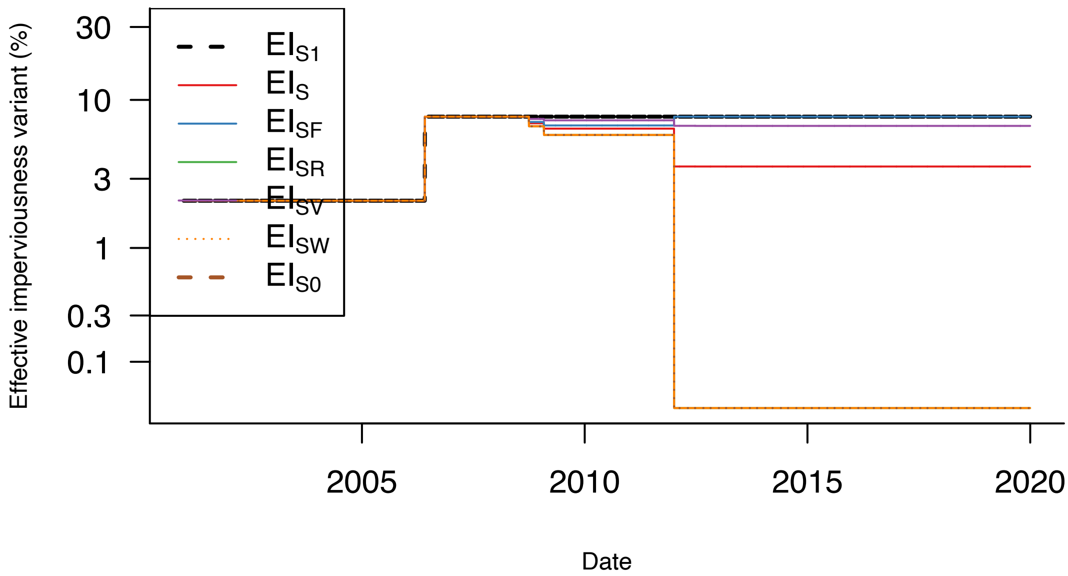

Fig. S4-4. Time series plot of six effective impervious variants for example subcatchment (Wattle Valley Rd North). Variant codes correspond to the EI subscripts used in the main paper.

```
load(here("data","lsc_rain_hourly.rda"))
lsc_rain_hourly$runoff_mm <- lose.init(lsc_rain_hourly$rain_mm,0.256,2)
lsc_runoff <- prepare_runoff_data(lsc_rain_hourly)
# system.time(
  budget_29 <- budget_subc_time_series(pipeID = 29, runoffData = lsc_runoff)
  #) #~ 2 min
layout(matrix(c(1,2),byrow = TRUE, 2,1), heights = c(11.5,12))
par(mar = c(2,4,1,1))
with(budget_29$daily, plot(date, ei_runoff/carea, type = 'l',
                          las = 1, ylab = "Daily runoff (mm/h)"));
with(budget_29$daily, lines(date, ei_scm_runoff/carea, col = 'red')); title(main = "a)", adj = 0)
par(mar = c(4,4,1,1))
with(budget_29$hourly[date > "2018-01-01"], plot(date, ei_runoff/carea, type = 'l',
                           las= 1, ylab = "Hourly runoff (mm/h)", xlab = "Date"))
with(budget_29$hourly[date > "2018-01-01"], lines(date, ei_scm_runoff/carea, col = 'red'));
title(main = "b)", adj = 0)
```

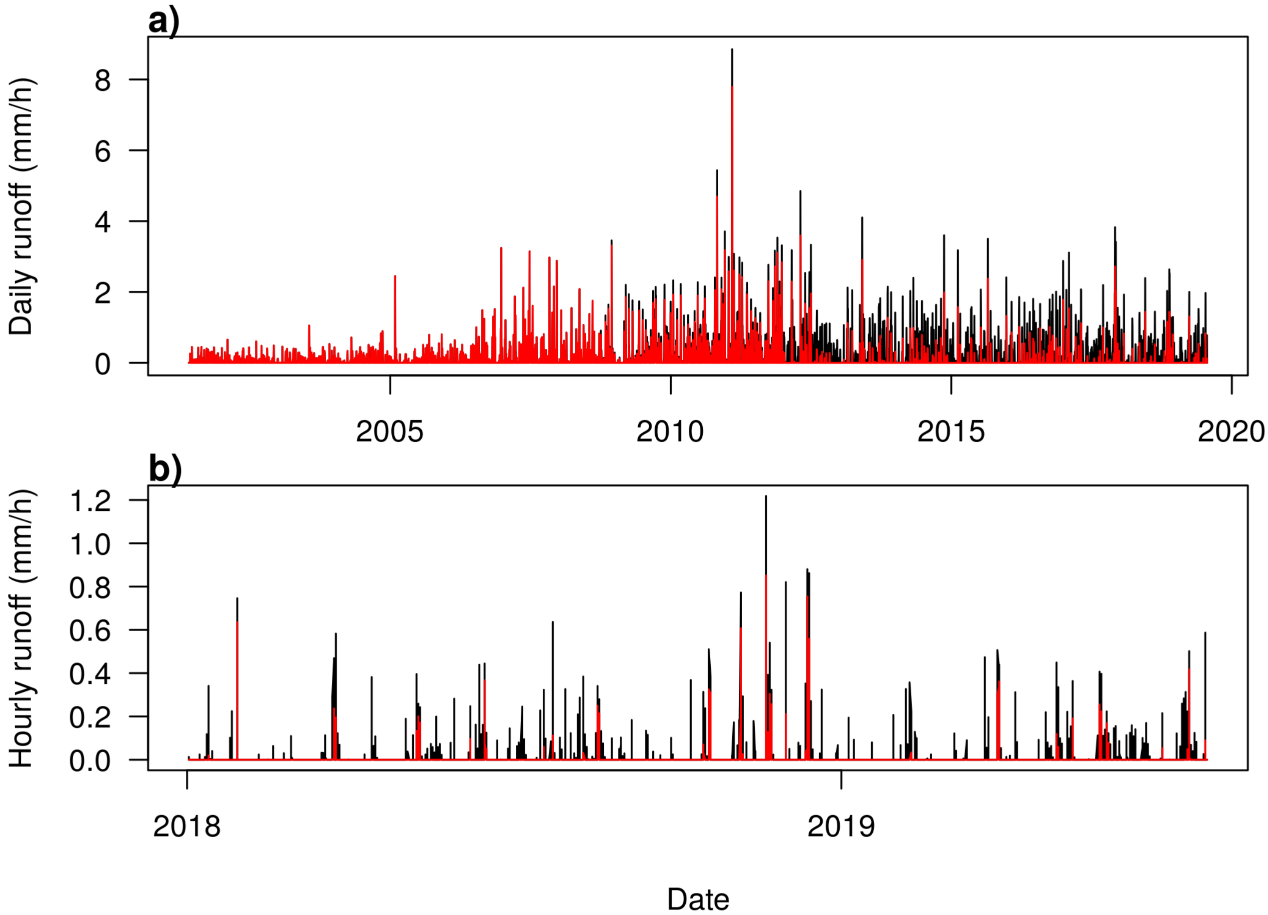

Fig. S4-5. Time series of modelled impervious runoff from example subcatchment (Wattle Valley Rd North): a) Daily flows over the full study period, b) Hourly flows for a detail of 1 year. Black lines show total impervious runoff and red lines show impervious runoff leaving the catchment after flowing through SCMs.

The non-linear trend in EI (Fig. S4-4) in this small catchment is because, prior to 2006, Wattle Valley Road was unsealed, and the impervious surfaces on properties of the subcatchment drained to the stream informally or along the roadside swale (and thus considered non-effective). In 2006, the road was sealed and curbed, with stormwater drainage connecting the road and houses of the subcatchment to the stream. SCMs installed in the project progressively reduced that connection, culminating in the construction of a large raingarden at the bottom of the subcatchment in 2012, which captured runoff from all subcatchment impervious surfaces.

Fig. S4-5a shows the models assume no impervious runoff from the subcatchment before sealing and piping of Wattle Valley Road in 2006, and a reduction in impervious runoff leaving the catchment following installation of SCMs in the subcatchmnent. Fig. S4-5a, shows that the frequency of overflow from the terminal SCMs in this subcatchment was reduced to ~12 rain events per year, which is sufficient for the runoff frequency sub-index (ESR) to achieve a perfect score, i.e. E_{SR} = 1 in Fig. 5a. Conversely, volumes exfiltrated from the terminal raingardens exceeded the target range, resulting in S_F equalling 1, and E_{SF} equalling E_{S1}.

Logic and use of the EI_subc_time_series and budget_subc_time_series functions

For every SCM, a budget of water fluxes is calculated using a time series of rainfall, impervious runoff and potential evapotransipiration, prepared using the `prepare_runoff_data` function (Fig. S4-2): for EB calculations this series is the standard 1965-1966 Croydon rainfall year (see S3), while for budget calculations it is the observed rainfall during the period of SCM changes being modelled. The three types of SCM have separate functions for calculating budgets. calcRGBudget and calcTankBudget adapt the gardenmodel and tankmodel functions, respectively (in modelfunctions2017.R, used by the EB calculator, https://tools.thewerg.unimelb.edu.au/EBcalc/, [3]), to permit specification parameters to be read from the database tables, "tanks" and "raingardens." calcDDBudget uses a simple look-up table to model the largely ineffective downpipe diverters that were installed in the Dobsons Creek catchment.

The budget_scm_on_datex function calculates a time series of water fluxes (i.e. a budget) into and out of an SCM (specified by scmID) given its specifications (including inflows from upstream SCMs) on a specified date (datex). The appropriate input data are compiled using data_on_datex. This function, subsets the database data to those elements of relevance to scmID, and adjusts specifications for datex, including ensuring that the network of SCMs upstream of scmID (if any) are correctly specified through the "nextds" field of the SCMs table. The function then uses the three calc_..._Budget functions as needed to calculate budgets for all upstream SCMs to ensure correct inflow into scmID, and finally calculates the budget for scmID on datex.

The time series of the budget produced by budget_scm_on_datex depends on whether the output is to be used to calculate EB (and therefore EI variants) or to be used to calculate the budget for a specified time period (a decision specified by the "specs_for_EB" argument). If EB is to be calculated, the function uses the 1-year Croydon 1965-1966 rainfall time series (and each SCM is assumed to be half-full at the start of the time series). The function EB_scm_on_datex takes this output and calculates EB index and its sub-indices for scmID on datex.

If the required output is a budget, then the output budget time series matches the time series of the input runoff data, which can be of any length of at least 1 day. This is to permit calculation of budgets in periods between changes in SCM specifications: in such cases, the starting volume of store in each SCM equals the final store volume in the previous run. Such a series of SCM budget calculations is calculated with the function budget_scm_time_series. This function uses data_for_scm to determine all the dates on which specifications relevant to scmID occurred, and runs budget_scm_on_datex in a loop for each date to build up an hourly series of flows into and out of the scm and daily time series of all elements of the water budget for the scm.

If a time series of EB statistics is required, then EB_scm_time_series works through a similar process of identifying change dates of relevance and uses EB_scm_on_datex to calculate EB statistics on each change date.

The ia_ts function is used to serve as a template output for the two _subc_time_series functions: a time series of total impervious area (tia) and effective impervious area (eia) for the nominated subcathment (pipeID).

EI_subc_time_series and budget_subc_time_series both start by identifying all dates on which terminal SCMs (those SCMs with no other SCMs downstream) changed in pipeID. For each such date, EI_subc_time_series calculates the EB statistics for each terminal SCM and divides each index by the maximum potential EB for that SCM (impervious area x 100) to get the EIA (effective impervious area) variant for that SCM. EIA variants for each date are calculated by summing EIA for all terminal SCMs and the subcatchment effective impervious area not treated by any SCM. EI is then calculated for each date by dividing by the subcatchment area. The statistics for each day are added to the output from ia_ts (e.g. Fig. 4).

For each change date, budget_subc_time_series uses budget_scm_time_series to calculate a time series of water fluxes from each terminal SCM for the period until the next change date.

It loops through each SCM, subtracting the impervious area treated by the SCM at each time step from the eia as estimated by `ia_ts`, and adding the time series of flow components passing through each SCMs together. Once budgets for all terminal SCMs have been calculated, impervious runoff from the untreated impervious surfaces of the subcatchment is added to outflows from the SCMs to produce two flow series (total impervious runoff, ei_runoff, and impervious runoff reaching the subcatchment outlet having passed through SCMs, ei_scm_runoff, see Fig. S4-5).

References

1. R Core Team. R: A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing; 2020. Available: https://www.R-project.org/

2. Walsh CJ, Burns MJ, Fletcher TD, Bos DG, Kunapo J, Poelsma P, et al. Linking stormwater control performance to stream ecosystem outcomes: Incorporating a performance metric into effective imperviousness/data and code. 2021. Available: https://osf.io/57azq

3. Fletcher TD, Walsh CJ, Bos D, Nemes V, RossRakesh S, Prosser T, et al. Restoration of stormwater retention capacity at the allotment-scale through a novel economic instrument. Water Science and Technology. 2011;64.2: 494–502. doi:10.2166/wst.2011.184