Reproducible code for manuscript figure 05 – Temporal Monte Carlo output

J.A. Torres-Matallana

2020-06-28

```
# organization 1: Luxembourg Institute of Science and Technology (LIST), Belvaux, Luxembourg
# organization 2: Wagenigen University and Research Centre (WUR), Wageningen, The Netherlands

# date: 27.06.2020 - 28.06.2020
```

Compile Rmarkdown file

```
library("rmarkdown")
rmarkdown::render("Main_reproducible_MC_outcomes.R")
```

Setup

```
library(knitr)
library(xts)
library(ggplot2)
library(gridExtra)
library(tikzDevice)

Sys.setenv("LANGUAGE"="En")
Sys.setlocale("LC_ALL", "en_GB.UTF-8")
```

```
## [1] "LC_CTYPE=en_GB.UTF-8;LC_NUMERIC=C;LC_TIME=en_GB.UTF-8;LC_COLLATE=en_GB.UTF-8;LC_MONETARY=en_GB.U
```

```
timing.ini <- Sys.time()
```

Loading functions

Loading data

```
load("data/data_mc_output.RData")
```

Plot functions

Function to create plot

```
My.pot.plot <- function(data1, xlab, ylab, y.intercept1, y.intercept2,
                        breaks, breaks.minor, labels, xlim){
  data1.xts <- xts(data1, order.by = data1[,1])
  y.max <- max(as.numeric(data1.xts[paste0(xlim[1], "/", xlim[2])]$q995))
  plot1 <- ggplot(data1, aes(x = Index, y = q995)) +
```

```
    theme_bw() +
    scale_x_datetime(name = xlab, breaks = breaks, labels = labels, limits = xlim,
                     minor_breaks = breaks.minor) +
    ylim(0, y.max*1.10) +
    theme(plot.margin = unit(c(1,12,5,6), units="points"),
          axis.text.x = element_text(angle = 90, vjust = 0.5,  hjust = 0),
          text = element_text(size=22)) +
    geom_ribbon(aes(ymin=q05,ymax=q95), fill="grey90", alpha=1) +
    geom_ribbon(aes(ymin=q005,ymax=q995), fill="grey70", alpha=.3) +
    geom_line(aes(y = mean),lwd = 0.5, colour = "blue") +
    geom_hline(yintercept=y.intercept1, linetype="dotted", color = "red") +
    geom_hline(yintercept=y.intercept2, linetype="dashed", color = "red") +
    #geom_line(aes(y = q05, colour = "grey"), size = 0.05)+
    #geom_line(aes(y = q95, colour = "grey"), size = 0.05)+
    labs(y = ylab, x = xlab)

  return(plot1)
}
```

Function to compose plot

```
My.plot.composed <- function(time.plot.ini, time.plot.end, breaks.by,
                             ylim_p, ylim1, ylim2, ylim3, ylim4, ylim5){
  breaks <- seq.POSIXt(from = time.plot.ini, by = breaks.by, to = time.plot.end)
  breaks.minor <- seq.POSIXt(from = time.plot.ini, by = 60*60*1, to = time.plot.end)
  labels <- format(breaks, "%b %d %H:%M")
  xlim <- c(time.plot.ini, time.plot.end)

  (plot_p <- My.pot.plot(data1 = data.p, xlab = "", ylab = "P [mm]",
                         y.intercept1 = NULL, y.intercept2 = NULL,
                         breaks = breaks, breaks.minor = breaks.minor, labels = labels,
                         xlim = xlim
  ))
  plot_p <- plot_p + theme(axis.text.x = element_blank())

  (plot1 <- My.pot.plot(data1 = data.qsv, xlab = "", ylab = "Qsv [l/s]",
                        y.intercept1 = 37.5, y.intercept2 = 75,
                        breaks = breaks, breaks.minor = breaks.minor, labels = labels,
                        xlim = xlim
  ))
  plot1 <- plot1 + theme(axis.text.x = element_blank())

  (plot2 <- My.pot.plot(data1 = data.bcod, xlab = "", ylab = "BCOD,Sv [kg]",
                        y.intercept1 = NULL, y.intercept2 = NULL,
                        breaks = breaks, breaks.minor = breaks.minor, labels = labels,
                        xlim = xlim
  ))
  plot2 <- plot2 + theme(axis.text.x = element_blank())

  (plot3 <- My.pot.plot(data1 = data.bnh4, xlab = "", ylab = "BNH4,Sv [kg]",
                        y.intercept1 = NULL, y.intercept2 = NULL,
                        breaks = breaks, breaks.minor = breaks.minor, labels = labels,
                        xlim = xlim
  ))
```

```
  plot3 <- plot3 + theme(axis.text.x = element_blank())

  (plot4 <- My.pot.plot(data1 = data.ccod, xlab = "", ylab = "CCOD,Sv,av [mg/l]",
                        y.intercept1 = 90, y.intercept2 = 125,
                        breaks = breaks, breaks.minor = breaks.minor, labels = labels,
                        xlim = xlim
  ))
  plot4 <- plot4 + theme(axis.text.x = element_blank())

  (plot5 <- My.pot.plot(data1 = data.cnh4, xlab = "", ylab = "CNH4,Sv,av [mg/l]",
                        y.intercept1 = 2.5, y.intercept2 = 5,
                        breaks = breaks, breaks.minor = breaks.minor, labels = labels,
                        xlim = xlim
  ))


  # Get the gtables
  g_p <- ggplotGrob(plot_p)
  g1  <- ggplotGrob(plot1)
  g2  <- ggplotGrob(plot2)
  g3  <- ggplotGrob(plot3)
  g4  <- ggplotGrob(plot4)
  g5  <- ggplotGrob(plot5)

  # Set the widths
  g_p$widths <- g2$widths
  g3$widths <- g2$widths
  g4$widths <- g2$widths
  g5$widths <- g2$widths

  gp <- list(g_p, g1, g2, g3, g4, g5)

}
```

Plot graphs

Render latex file to pdf

Include pdf

Timing

```
timing.end <- Sys.time()
(timing.elapsed <- timing.end - timing.ini)
```

```
## Time difference of 8.087818 secs
```

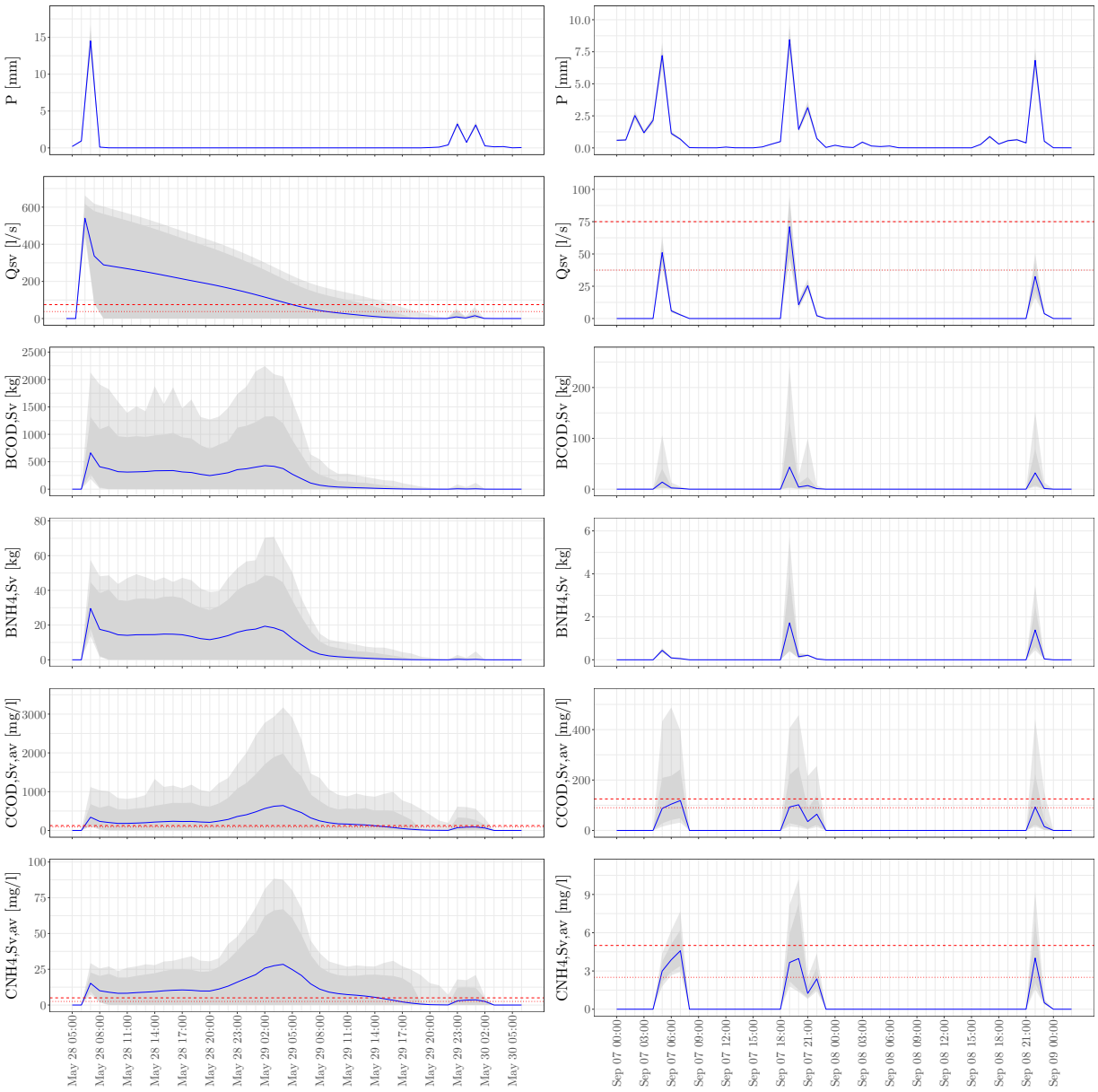

Figure 1: Figure 05.