The slow and steady salinization of Sparkling Lake

Code and Figures

by: Hilary Dugan, hdugan@wisc.edu

2020-11-29

The following code and figures supports the manuscript *The slow and steady salinization of Sparkling Lake* by Hilary Dugan and Linnea Rock. This code pulls data from the North Temperate Lakes Long-Term Ecological Research (NTL-LTER) core data sets from EDI. It then interpolates and decomposes chloride timeseries and builds a model of chloride concentration in Sparkling Lake.

Major ion chemistry in NTL-LTER lakes

Package ID: knb-lter-ntl.2.34 Cataloging System:https://pasta.edirepository.org. Data set title: North Temperate Lakes LTER: Chemical Limnology of Primary Study Lakes: Major Ions 1981 - current

```
inUrl2  <- "https://pasta.lternet.edu/package/data/eml/"
dataset <- "knb-lter-ntl/2/34/3f740d0b77b3caf6930a8ce9cca4306a"
infile2 <- tempfile()
download.file(paste0(inUrl2,dataset),infile2,method="curl")
LTERions <- read_csv(infile2, skip=1, quote ='"',guess_max = 20000, col_names=c(
    "lakeid","year4","daynum","sampledate","depth","rep","sta","event","cl",
    "so4","ca","mg","na","k","fe","mn","cond","flagcl","flagso4","flagca",
    "flagmg","flagna","flagk","flagfe","flagmn","flagcond"))
```

Lake Level of NTL-LTER lakes

Package ID: knb-lter-ntl.30.26 Cataloging System:https://pasta.edirepository.org. Data set title: North Temperate Lakes LTER: Lake Levels 1981 - current

```
inUrl2  <- "https://pasta.lternet.edu/package/data/eml/"
dataset <- "knb-lter-ntl/30/26/5f324be2082168798e8c83d693147fe0"
infile2 <- tempfile()
download.file(paste0(inUrl2,dataset),infile2,method="curl")
LTERlakelevel <- read_csv(infile2, skip=1, quote ='"',guess_max = 20000, col_names=c(
    "lakeid","year4","daynum","sampledate","sta","llevel_elevation")) %>%
    dplyr::filter(lakeid %in% c('SP','BM')) %>%
    group_by(lakeid) %>%
    mutate(rel.elev = scale(llevel_elevation, center = T, scale = F)[,1])
```

Lake characteristics and bathymetry NTL-LTER lakes

```
lakestats = read_csv('../data/LTERlakes.csv')
load('../data/NTLbathymetry_1m.rda')
```

Timeseries interpolation and decomposition

```
ions.wide = LTERions %>%
  dplyr::filter(lakeid %in% c('SP','BM')) %>%
  mutate(across(everything(), ~replace(., .<0 , NA))) %>%
  rename_at(vars(cl:cond), ~ str_c("value_",.)) %>%
  rename_at(vars(flagcl:flagcond), ~ str_c("error_",.)) %>%
  rename_all(~str_replace_all(.,"flag","")) %>%
  pivot_longer(-(lakeid:event), names_to = c('.value','item'), names_sep = '_') %>%
  filter(!is.na(value) & value>= 0) %>%
  filter(!str_detect(error,'A|K|L') | is.na(error)) # Remove suspect data
  # A sample suspect
  # L data and blind differ by more than 15%
  # K data suspect

decompose.list = list()
for (var in c('ca','mg','k','na','cl','so4')) {
  for (lakeAbr in c('SP','BM')) {
    maxdepth = lakestats %>% filter(LakeAbr == lakeAbr) %>%
      dplyr::select(SamplingDepth) %>% pull()

    iout = monthlyInterpolate(lakeAbr = lakeAbr, var = var, dfin = ions.wide,
                              maxdepth = maxdepth, printFigs = F)

    lakeBathy = NTLbathymetry_1m %>% dplyr::filter(LakeID == lakeAbr) %>%
      mutate(perArea = area/sum(area))

    df.area = iout$weeklyInterpolated %>%
      dplyr::arrange(date) %>%
      dplyr::left_join(lakeBathy) %>%
      dplyr::mutate(load = var * area) %>%
      dplyr::mutate(intConc = var * perArea) %>%
      as_tibble()

    df.load = df.area %>%
      group_by(date) %>%
      dplyr::summarise(load.kg = sum(load)/1000, intConc.mgL = sum(intConc)) %>%
      mutate(year = year(date), month = month(date))

    output = decomposeTS2(df.load = df.load, lakeAbr = lakeAbr, var = var) %>%
      dplyr::mutate(lakeid = lakeAbr, var = var)
    decompose.list[[paste0(lakeAbr,var)]] = output
  }
}
# unlist timeseries decomposition
decompose.df = bind_rows(lapply(decompose.list, as.data.frame.list))
```

Timeseries decomposition plots

```
# Seasonal Plots ####
labels2 <- as_labeller(c(ca = 'Ca^paste(2,"\u002B")', mg = 'Mg^paste(2,"\u002B")',
                         k = 'K^paste("\u002B")',
                         na = 'Na^paste("\u002B")', cl = 'Cl^paste("\u2212")',
```

```r
                         so4 = 'SO[4]^paste(2,"\u2212")'),
                       default = label_parsed)

p1 = ggplot(filter(decompose.df, decompose == 'var.seas')) +
  geom_path(aes(x = date, y = value, color = lakeid)) +
  scale_y_continuous(position = "right") +
  scale_color_manual(values = c('lightblue4','lightskyblue3')) +
  geom_vline(xintercept = seq.Date(from = as.Date('1980-01-01'),
                                   to = as.Date('2020-01-01'),by = 'year'),
             linetype=2, size = 0.2, alpha = 0.2) +
  facet_wrap(vars(var), scales = 'free', ncol = 1, strip.position = 'left',
             labeller = labels2) +

  xlim(as.Date('2010-01-01'), as.Date('2012-01-01')) +
  ylab('Seasonal Trend') +
  theme_bw(base_size = 8) +
  theme(legend.position = "none", axis.title.x = element_blank())


# Trends Plots ####
p2 = ggplot(decompose.df) +
  geom_line(data = filter(decompose.df, decompose == 'var.conc'), aes(x = date,
            y = value, color = lakeid),alpha = 0.5, size = 0.2) +
  geom_line(data = filter(decompose.df, decompose == 'var.trend'), aes(x = date,
            y = value, color = lakeid), size = 0.6) +
  scale_color_manual(values = c('lightblue4','lightskyblue3')) +
  geom_vline(xintercept = seq.Date(from = as.Date('1980-01-01'),
                                   to = as.Date('2020-01-01'),
            by = 'year'), linetype=2, size = 0.2, alpha = 0.2) +
  facet_wrap(vars(var), scales = 'free', ncol = 1, strip.position = 'right',
             labeller = labels2) +
  ylab(bquote('Depth-integrated Ion Concentration' ~ (mg~L^-1))) +
  theme_bw(base_size = 8) +
  theme(legend.position = "none", axis.title.x = element_blank())

# Lake level plots
p.ll = ggplot(filter(LTERlakelevel)) +
  geom_line(aes(x = sampledate, y = rel.elev, color = lakeid),
            size = 0.5) +
  geom_vline(xintercept = seq.Date(from = as.Date('1980-01-01'),
                                   to = as.Date('2020-01-01'),
            by = 'year'), linetype=2, size = 0.2, alpha = 0.2) +
  scale_color_manual(labels = c('Big Muskellunge','Sparkling'),
                     values = c('lightskyblue3','lightblue4'), name = 'Lake') +
  theme_bw(base_size = 8) +
  ylab('Relative Lake Level') +
  theme(axis.title.x = element_blank(), plot.title = element_blank())

p2 + p1 + p.ll + plot_annotation(tag_levels = 'a',tag_suffix = ')', caption =
'Figure2. a) Decomposed time-series of long-term trends (thick line) and observed concentrations
(thin line) in depth-integrated major ion concentrations in Big Muskellunge and Sparkling
Lakes. b) Two years of the decomposed seasonal component of time-series in a). c) Relative
lake-level fluctuations in meters. Each time-series was individually demeaned.') +
```

```
  plot_layout(heights = c(5, 0.8)) +
  guide_area() +
  plot_layout(guides = 'collect') &
  theme(plot.tag = element_text(size = 8),
        plot.title = element_text(size = 8, hjust = 0),
                                plot.caption = element_text(size = 8, hjust = 0))

ggsave('../output/Figure2_SP_BM_combo.png',width = 6, height = 6)
```

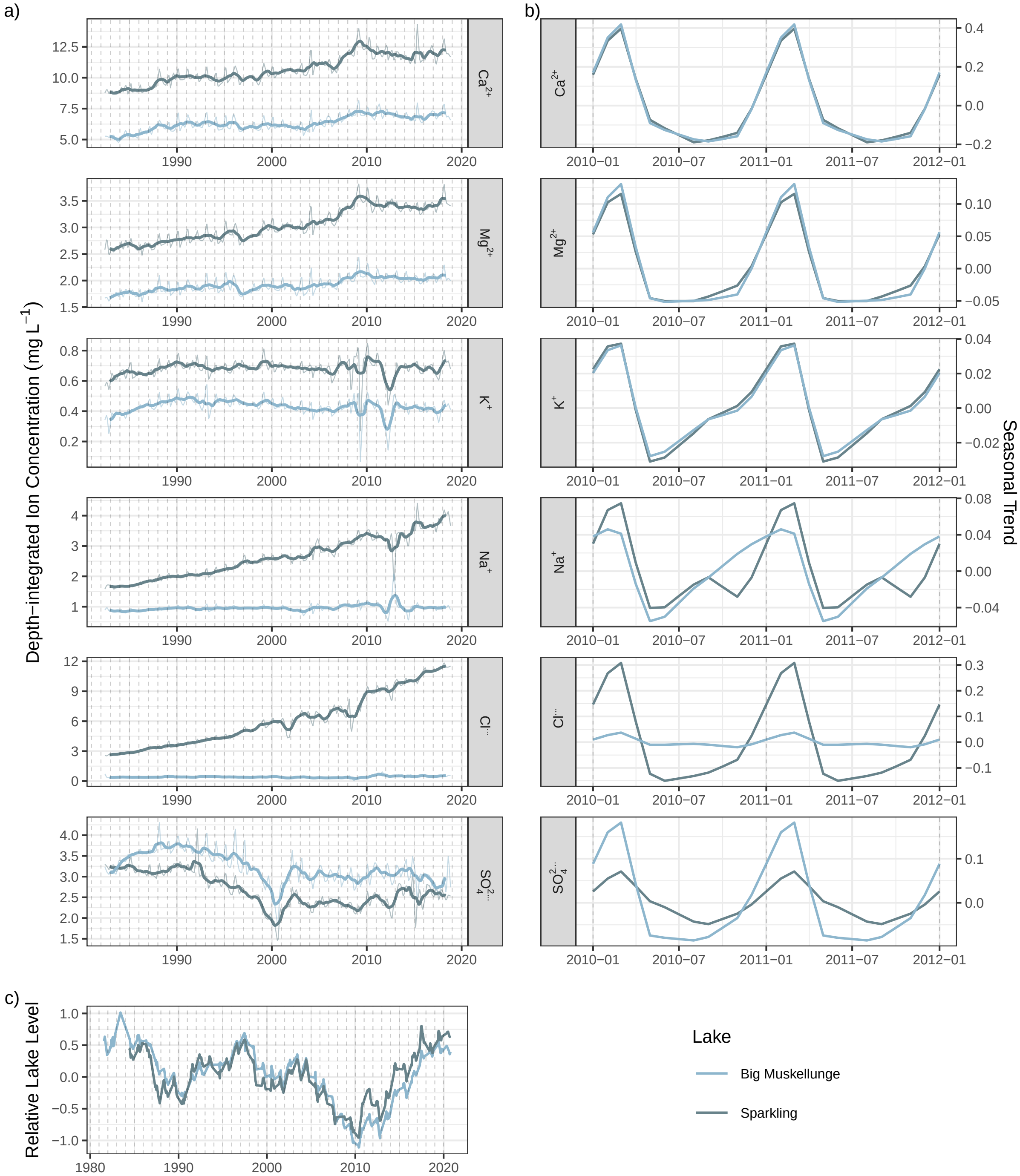

Figure2. a) Decomposed time−series of long−term trends (thick line) and observed concentrations (thin line) in depth−integrated major ion concentrations in Big Muskellunge and Sparkling Lakes. b) Two years of the decomposed seasonal component of time−series in a). c) Relative lake−level fluctuations in meters. Each time−series was individually demeaned.

Sparkling Lake chloride storage model

```r
sparkling = LTERions %>%
  dplyr::filter(lakeid == 'SP') %>%
  dplyr::filter(!is.na(cl)) %>%
  dplyr::select(sampledate:rep, cl, flagcl)

# Parameters drawn from literature
# Volume
volume = 637000 * 10.9 #m3
# alpha = QGo (annual flux of groundwater out) / Volume of lake
alphat = 4.2e5/volume
# Krabbenhoft, GWin = 29 cm/yr, PrecipIn = 79 cm/yr
Cinf = ((29/108)*68) + ((79/108)*0.1)

#### Find values for initial storage and B
optim(par = c(0.03,0.075), fn = getfit, method = 'Nelder-Mead')
```

```
## $par
## [1] 0.02494550 0.07867549
##
## $value
## [1] 0.2256854
##
## $counts
## function gradient
##       71       NA
##
## $convergence
## [1] 0
##
## $message
## NULL
```

```r
# Use optim parameters for y and B
C = 1
C.bowser = 1
y = 0.025 # Initial storage used from optim
S = 1-y #
B = 0.07868 # From optim
alphat = 4.2e5/volume
for (t in 2:131){
  y[t] = y[t-1] * exp(B) / (1 - y[t-1] + (y[t-1] * exp(B)))
  S[t] = 1 - y[t]

  # C[t] = Cinf + (C[t-1] - Cinf) * exp(-alphat)
  C.storage.inf = Cinf * y[t]
  C[t] = C.storage.inf + ((C[t-1] - C.storage.inf) * exp(-alphat)) # with storage
  C.bowser[t] = Cinf + (C.bowser[t-1] - Cinf) * exp(-alphat)
}

df = data.frame(year = 1950:2080, C = C, C.bowser = C.bowser, y = y, S = S)
```

```
p2 = ggplot(df) + geom_line(aes(x = year, y = C), color = 'black', size = 0.9) +
  geom_line(aes(x = year, y = C.bowser), color = 'lightblue4', size = 0.7,
            alpha = 0.7, linetype = 2) + # Bowser Model
  ylab(bquote('Sparkling Lake Chloride' ~ (mg~L^-1))) +
  geom_point(data = sparkling, aes(x = year(sampledate), y = cl), shape = 21,
             fill = 'lightblue2', size = 0.9) + # Sparkling Lake observations
  geom_line(aes(x = year, y = y*5), color = 'gray50', size = 0.7, linetype = 3) + # Soil storage
  scale_y_continuous(expand = c(0,0), sec.axis = sec_axis(trans = ~0.2*.,
         name =bquote('Storage capacity of soil' ~ (S[t])),
       breaks = c(0.5,1), labels = c(0.5,1))) + #scaled second axis +
  theme_bw(base_size = 8) +
  theme(axis.title.x = element_blank(), plot.title = element_blank()) +
  theme(axis.title.y.right = element_text(hjust=c(1))) +
  NULL

p2 +
  plot_annotation(#tag_levels = 'a',tag_suffix = ')',
  caption =
'Figure 3. Chloride concentrations in Sparkling Lake. Blue points are measured observations, the
dashed line is the original box model from Bowser (1990),  the black line is the soil-reservoir
box model, and the dotted line is the used proportion of soil storage capacity.',
   theme = theme(plot.tag = element_text(size = 8),
                              plot.title = element_text(size = 8, hjust = 0),
                              plot.caption = element_text(size = 8, hjust = 0)))
ggsave('../output/Figure3_StorageModel_caption.png', width = 6, height = 2.5, dpi = 500)
```

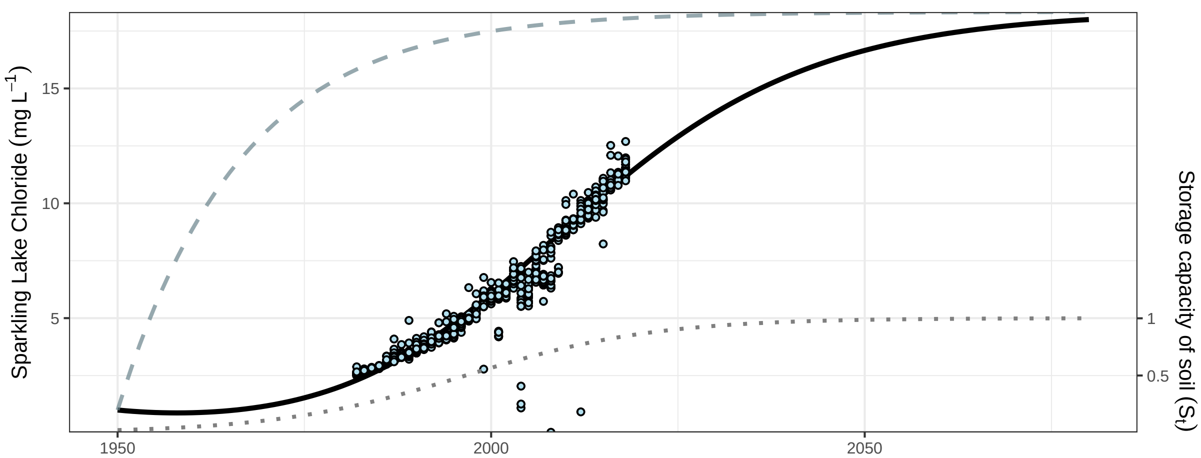

Figure 3. Chloride concentrations in Sparkling Lake. Blue points are measured observations, the dashed line is the original box model from Bowser (1990), the black line is the soil−reservoir box model, and the dotted line is the used proportion of soil storage capacity.