Identify control chronologies for detecting defoliation by the western spruce budworm in Douglas-fir radial growth time series based on climate and tree-ring data for: Drought may initiate western spruce budworm outbreaks, but multi-year periods of increased moisture availability promote widespread defoliation

Sarah J. Hart, Olivia Santiago, Josh Carrell, and Thomas T. Veblen

February 04, 2025

# Overview

Briefly, here we matched nonhost sites with host sites based the highest Pearson’s correlation, subject to the following constraints:

1. the nonhost chronology overlapped the entire host chronology
2. the nonhost chronology was less than 125 km away from the host site.

Finally, we removed any sites where the host chronology was poorly correlated (r<0.2) with either the SC-PDSI or the available nonhost chronologies. Results of this matching are presented in Table ).

host.meta <- read.csv(here("Data", "TreeRing", "Processed", "Host", "host-metadata.csv"))  
nonhost.meta <- read.csv(here("Data", "TreeRing", "Processed", "Nonhost", "nonhost-metadata.csv")) %>% filter(is.na(Lat)==F)  
proj.proj <- 4629  
  
nonhost.sites <- st\_as\_sf(x = nonhost.meta, coords = c("Lon", "Lat"), crs = "+proj=longlat +ellps=WGS84 +datum=WGS84 +no\_defs") %>% st\_transform(proj.proj)  
  
host.sites <- st\_as\_sf(x = host.meta, coords = c("Lon", "Lat"), crs = "+proj=longlat +ellps=WGS84 +datum=WGS84 +no\_defs") %>% st\_transform(proj.proj)

# Read in NADA time series  
nada.hosts <- read.csv(here("Data", "nada-hosts.csv"))

### READ IN RINGWIDTH DATA  
# Host data  
host.rwls <- list.files(here("Data", "TreeRing", "Processed", "Host"), pattern=".rwl", full.names=T)  
names(host.rwls) <- gsub(".rwl", "", unlist(list.files(here("Data", "TreeRing", "Processed", "Host"), pattern=".rwl", full.names=F)))  
host.rwls <- lapply(host.rwls, read.rwl)  
  
# Nonhost data  
nonhost.rwls <- list.files(here("Data", "TreeRing", "Processed", "NonHost"), full.names = T, pattern=".rwl")  
names(nonhost.rwls) <- gsub(".rwl", "", unlist(list.files(here("Data", "TreeRing", "Processed", "NonHost"), pattern=".rwl", full.names=F)))  
nonhost.rwls <- lapply(nonhost.rwls, read.rwl)  
namez <- names(nonhost.rwls)   
bad.namez <- which(namez %in% c("CO532", "CO598", "DRI", "DRIre", "CO534", "ELDOre", "CO539", "CO608", "JEFFre", "CO030", "CO541", "CO625", "CO653", "CO548", "CO675", "CO550", "CO641", "VANBre"))  
nonhost.rwls <- nonhost.rwls[-bad.namez]  
  
### DETREND ###  
# Host data  
host.rwis <- lapply(host.rwls, detrend, method = "ModNegExp")  
host.rwis <- lapply(host.rwis, detrend, method = "Spline", nyrs = 30)  
  
# Nonhost data  
nonhost.rwis <- lapply(nonhost.rwls, detrend, method = "ModNegExp")  
nonhost.rwis <- lapply(nonhost.rwis, detrend, method = "Spline", nyrs = 30)  
  
### BUILD CHRONOLOGIES  
# Host data  
host.crns <- lapply(host.rwis, chron, prewhiten = TRUE, prefix="")  
host.crns <- lapply(host.crns, "[",c(-1, -3))  
host.crns.ts <- lapply(host.crns, FUN=function(x){return( ts(x, start=min(as.numeric(row.names(x)))) )})  
host.crns.mts<- do.call(cbind, host.crns.ts)  
  
# Nonhost data  
nonhost.crns <- lapply(nonhost.rwis, chron, prewhiten = TRUE, prefix="")  
nonhost.crns <- lapply(nonhost.crns, "[",c(-1, -3))  
nonhost.crns.ts <- lapply(nonhost.crns, FUN=function(x){return( ts(x, start=min(as.numeric(row.names(x)))))})  
nonhost.crns.mts<- do.call(cbind, nonhost.crns.ts)  
  
## CALCULATE DISTANCES  
host.pts <- st\_as\_sf(x = host.meta, coords = c("Lon", "Lat"), crs = "+proj=longlat +ellps=WGS84 +datum=WGS84 +no\_defs") %>% st\_transform(proj.proj)  
  
nonhost.pts <- st\_as\_sf(x = nonhost.meta[!is.na(nonhost.meta$Lat)==T,], coords = c("Lon", "Lat"), crs = "+proj=longlat +ellps=WGS84 +datum=WGS84 +no\_defs") %>% st\_transform(32613) %>% filter(SeriesCode %in% names(nonhost.crns)) %>% st\_transform(proj.proj)  
  
dists <- st\_distance(host.pts, nonhost.pts, by\_element=F)   
row.names(dists)<- host.pts$SeriesCode  
colnames(dists)<- nonhost.pts$SeriesCode  
  
### CALCUALTE CORRELATIONS  
cor.mat <- cor(cbind(host.crns.mts, nonhost.crns.mts), use="pairwise.complete.obs")  
row.names(cor.mat) <- c(colnames(host.crns.mts), colnames(nonhost.crns.mts))  
colnames(cor.mat) <- c(colnames(host.crns.mts), colnames(nonhost.crns.mts))  
  
# Summarize  
res <- data.frame(host=host.meta$SeriesCode, nonhost1=NA, nonhost2=NA, nonhost3=NA, nonhost1.r=NA, nonhost2.r=NA, nonhost3.r=NA, nonhost1.d=NA, nonhost2.d=NA, nonhost3.d=NA, r.hostxNADA=NA) # create data frame to hold results  
for(j in res$host){  
   
 # (1) limit to chronologies that overlap the entire host record  
 lastyrj <- host.meta %>% dplyr::filter(SeriesCode == j) %>% pull(LastYear)  
 firstyrj <- host.meta %>% dplyr::filter(SeriesCode == j) %>% pull(FirstYear)  
 firstyrj <- ifelse(firstyrj<1650, 1650, firstyrj)  
 nonhost.long <- nonhost.meta %>% filter(SeriesCode %in% nonhost.pts$SeriesCode & LastYear>=lastyrj & FirstYear<=firstyrj) %>% pull(SeriesCode)  
   
 # (2) limit to sites within 100 km   
 dists.j <- as.data.frame(dists)  
 dists.j <- dists[j, nonhost.long]  
 dists.j <- unclass(dists.j)  
 dists.j <- dists.j[dists.j <= 125000]   
   
 # (3) determine top 3 correlations  
 cor.matj <- cor.mat[j, names(dists.j)]  
 top3 <- sort(cor.mat[j, names(dists.j)],decreasing=T)[1:3]  
 res[res$host==j, c("nonhost1", "nonhost2", "nonhost3")] <- names(top3)  
 res[res$host==j, c("nonhost1.r", "nonhost2.r", "nonhost3.r")] <- cor.matj[names(top3)]  
 res[res$host==j, c("nonhost1.d", "nonhost2.d", "nonhost3.d")] <- dists.j[names(top3)]  
   
 # (4) correlation between host and NADA  
 nada.ts <- ts(nada.hosts[,j], start=min(nada.hosts$X))  
 res[res$host==j, ]$r.hostxNADA <- cor(cbind(host.crns.mts[,colnames(host.crns.mts)==j], nada.ts), use="pairwise.complete.obs")[1,2]  
  
}  
res$mean.r <- res %>% dplyr::select(nonhost1.r:nonhost3.r) %>% rowMeans(na.rm=T)  
res$notes <- NA  
res2 <- res  
  
res2 <- res2 %>% filter(r.hostxNADA>=0.2) # Estes park, Frankenberger Point, or South Hollowell do not correlate well with PDSI  
res <- res %>% mutate(notes=ifelse(r.hostxNADA<0.2, "Removed due to poor correlation with PDSI", NA))  
  
write.csv(res, here("Results", "hostXnonhost.csv"), row.names=F)  
write.csv(res2, here("Results", "hostXnonhost-subset.csv"), row.names=F)

res <- read.csv(here("Results", "hostXnonhost.csv"))  
res <- res %>% mutate(nonhost1.r=round(nonhost1.r,digits=2), nonhost2.r=round(nonhost2.r,digits=2), nonhost3.r=round(nonhost3.r,digits=2), nonhost1.d=round(nonhost1.d/1000,digits=0), nonhost2.d=round(nonhost2.d/1000,digits=0), nonhost3.d=round(nonhost3.d/1000,digits=0), r.hostxNADA=round(r.hostxNADA,digits=2))  
  
res <- res %>% unite("Control1", nonhost1, nonhost1.r, remove = TRUE, sep=" (r=") %>% mutate(Control1=paste0(Control1, "; ")) %>% unite("Control1", Control1, nonhost1.d, remove = TRUE, sep="d=") %>% mutate(Control1=paste0(Control1, " km)"))   
  
res <- res %>% unite("Control2", nonhost2, nonhost2.r, remove = TRUE, sep=" (r=") %>% mutate(Control2=paste0(Control2, "; ")) %>% unite("Control2", Control2, nonhost2.d, remove = TRUE, sep="d=") %>% mutate(Control2=paste0(Control2, " km)"))   
  
res <- res %>% unite("Control3", nonhost3, nonhost3.r, remove = TRUE, sep=" (r=") %>% mutate(Control3=paste0(Control3, "; ")) %>% unite("Control3", Control3, nonhost3.d, remove = TRUE, sep="d=") %>% mutate(Control3=paste0(Control3, " km)"))   
   
res <- res %>% dplyr::select(host, Control1, Control2, Control3, r.hostxNADA, notes)  
colnames(res) <- c("Host site", "Nonhost control site 1", "Nonhost control site 2", "Nonhost control site 3", "Correlation with PDSI", "Notes")  
  
res %>% flextable() %>% flextable::align(j=-1, align = "center", part = "all") %>% set\_table\_properties(layout = "autofit", width=1)

**Table** **:** Summary of nonhost chronologies matched with each host site. For the top three nonhost sites, the Pearson's correlation coefficient (r) and distance (d) between the host and nonhost sits are displayed in parentheses below the name of each series.

| Host site | Nonhost control site 1 | Nonhost control site 2 | Nonhost control site 3 | Correlation with PDSI | Notes |
| --- | --- | --- | --- | --- | --- |
| TI | CO591 (r=0.34; d=94 km) | DRIcombo (r=0.28; d=109 km) | CO622 (r=0.24; d=99 km) | 0.25 |  |
| B18 | CO666 (r=0.54; d=10 km) | CO607 (r=0.54; d=5 km) | CO611 (r=0.46; d=70 km) | 0.38 |  |
| B19 | CO607 (r=0.51; d=7 km) | CO666 (r=0.41; d=10 km) | CO611 (r=0.34; d=71 km) | 0.33 |  |
| EP | CO601 (r=0.09; d=112 km) | CO622 (r=0.04; d=42 km) | JEFFcombo (r=0.04; d=79 km) | 0.11 | Removed due to poor correlation with PDSI |
| FP | CO639 (r=0.17; d=114 km) | CO666 (r=0.12; d=23 km) | CO601 (r=0.12; d=77 km) | 0.14 | Removed due to poor correlation with PDSI |
| LJ | CO666 (r=0.47; d=5 km) | CO607 (r=0.46; d=18 km) | CO611 (r=0.43; d=84 km) | 0.47 |  |
| SS | CO607 (r=0.41; d=11 km) | CO666 (r=0.34; d=24 km) | CO639 (r=0.32; d=113 km) | 0.30 |  |
| WB | CO607 (r=0.34; d=14 km) | CO666 (r=0.33; d=11 km) | CO591 (r=0.31; d=86 km) | 0.38 |  |
| NI | CO591 (r=0.3; d=78 km) | CO622 (r=0.25; d=52 km) | CO596 (r=0.2; d=56 km) | 0.28 |  |
| SP | CO622 (r=0.38; d=54 km) | CO611 (r=0.36; d=91 km) | CO607 (r=0.35; d=35 km) | 0.53 |  |
| SH | CO601 (r=0.07; d=113 km) | VANBcombo (r=0.05; d=31 km) | CO596 (r=0.03; d=53 km) | 0.03 | Removed due to poor correlation with PDSI |
| SR | CO601 (r=0.46; d=28 km) | CO602 (r=0.46; d=48 km) | CO611 (r=0.45; d=48 km) | 0.39 |  |
| JP | CO639 (r=0.54; d=8 km) | CO669 (r=0.39; d=113 km) | TCcombo (r=0.37; d=70 km) | 0.38 |  |
| WR | CO639 (r=0.58; d=9 km) | CO601 (r=0.51; d=38 km) | CO602 (r=0.51; d=25 km) | 0.50 |  |
| WW | CO639 (r=0.43; d=26 km) | CO601 (r=0.39; d=18 km) | CO602 (r=0.39; d=43 km) | 0.28 |  |