Prevision of comparisons with the ALFAM2 model

Sasha D. Hafner

16 May, 2025

Overview

The point of this document is to look at precision of the ALFAM2 model when comparing emission between two similar conditions.

1. Prep

```r
library(ALFAM2)
library(data.table)
library(ggplot2)
packageVersion('ALFAM2')
```

```
## [1] '4.2.8'
```

Read input data.

```r
dat <- fread('input.csv')
```

Add other variables.

```r
dat[, `:=` (app.rate.ni = 25, man.dm = 4.5, man.ph = 7.2)]
dat[, rain.rate := precipitation_mm / 1]
dat[, air.temp := temperature_celsius]
dat[, wind.2m := windspeed_m_s]
dat[, wind.sqrt := sqrt(windspeed_m_s)]
```

Get two subsets, with application starting at 12:00 and 20:00 on the same day, with emission predicted over 48 hours.

```r
d1 <- dat[7:(7 + 48), ]
d2 <- dat[15:(15 + 48), ]
```

Get time in hours.

```r
d1[, ct := as.numeric(difftime(time, min(time), units = 'hours'))]
d2[, ct := as.numeric(difftime(time, min(time), units = 'hours'))]
```

Take a look at the tops.

```r
head(d1[, .(time, ct, app.rate.ni, man.dm, man.ph, air.temp, wind.2m)])
```

```
##                    time    ct app.rate.ni man.dm man.ph air.temp  wind.2m
##                  <POSc> <num>       <num>  <num>  <num>    <num>    <num>
## 1: 2025-05-15 12:00:00     0          25    4.5    7.2 18.75641 2.511143
## 2: 2025-05-15 13:00:00     1          25    4.5    7.2 19.38350 2.757864
```

```
## 3: 2025-05-15 14:00:00        2          25     4.5      7.2 19.68736 3.198003
## 4: 2025-05-15 15:00:00        3          25     4.5      7.2 19.54932 3.449661
## 5: 2025-05-15 16:00:00        4          25     4.5      7.2 18.98815 3.441642
## 6: 2025-05-15 17:00:00        5          25     4.5      7.2 18.01563 3.444468
```

```
head(d2[, .(time, ct, app.rate.ni, man.dm, man.ph, air.temp, wind.2m)])
```

```
##                      time   ct app.rate.ni man.dm man.ph  air.temp  wind.2m
##                    <POSc> <num>      <num> <num> <num>      <num>    <num>
## 1: 2025-05-15 20:00:00      0          25     4.5      7.2 13.969246 3.354596
## 2: 2025-05-15 21:00:00      1          25     4.5      7.2 12.419703 3.457194
## 3: 2025-05-15 22:00:00      2          25     4.5      7.2 11.261217 3.697640
## 4: 2025-05-15 23:00:00      3          25     4.5      7.2 10.345874 3.697640
## 5: 2025-05-16 00:00:00      4          25     4.5      7.2  9.420493 3.717327
## 6: 2025-05-16 01:00:00      5          25     4.5      7.2  8.428627 3.683218
```

Note that these inputs are different from the values shown in the screenshot in the original email message with the question!

Combine into one.

```
d1[, apptime := 'T12']
d2[, apptime := 'T20']

din <- rbind(d1, d2)
```

2. Compare parameter sets 2 and 3

Run ALFAM2 model with parameter sets 2 and 3, combine those results, and get the final values for a comparison.

```
pred2 <- alfam2(din, pars = alfam2pars02, group = 'apptime')
```

```
## Warning in alfam2(din, pars = alfam2pars02, group = "apptime"): Argument app.name is missing or dat
##    So function will return relative emission only.
```

```
## User-supplied parameters are being used.
```

```
## Warning in prepDat(dat, warn = warn): Argument prep.dum = TRUE but there are no variables to convert
##    Ignoring prep.dum = TRUE.
```

```
## Warning in alfam2(din, pars = alfam2pars02, group = "apptime"): Incorporation columns  were dropped
##     because argument time.incorp is NULL
##     So there is no incorporation.
##     Set check = FALSE to not drop, but then check output.
```

```
## Warning in alfam2(din, pars = alfam2pars02, group = "apptime"): Running with 12 parameters. Dropped
## These secondary parameters have been dropped:
##    app.mthd.os.f0
##    man.source.pig.f0
##    app.mthd.cs.f0
##    app.mthd.bc.r1
##    app.mthd.ts.r1
##    ts.cereal.hght.r1
##    app.mthd.bc.r3
##    app.mthd.cs.r3
##    incorp.shallow.f4
##    incorp.shallow.r3
```

```
##    incorp.deep.f4
##    incorp.deep.r3

pred3 <- alfam2(din, pars = alfam2pars03, group = 'apptime', conf.int = 0.9)

## Warning in alfam2(dat = dat, pars = pars, add.pars = add.pars, app.name = app.name, : Argument app.na
##      So function will return relative emission only.

## Default parameters (Set 3) are being used.

## Warning in prepDat(dat, warn = warn): Argument prep.dum = TRUE but there are no variables to convert
##    Ignoring prep.dum = TRUE.

## Warning in alfam2(dat = dat, pars = pars, add.pars = add.pars, app.name = app.name, : Incorporation
##      because argument time.incorp is NULL
##      So there is no incorporation.
##      Set check = FALSE to not drop, but then check output.

## Warning in alfam2(dat = dat, pars = pars, add.pars = add.pars, app.name = app.name, : Running with 1
## These secondary parameters have been dropped:
##    app.mthd.os.f0
##    app.mthd.cs.f0
##    man.source.pig.f0
##    app.mthd.bc.r1
##    app.mthd.ts.r1
##    app.mthd.cs.r3
##    incorp.deep.r3
##    incorp.shallow.f4
##    incorp.deep.f4
```

```r
setDT(pred2)
setDT(pred3)

pred2[, parset := 'pars 2']
pred3[, parset := 'pars 3']

preds <- rbind(pred2, pred3, fill = TRUE)

pend <- preds[ct == 48, ]
pend <- dcast(pend, parset ~ apptime, value.var = 'er')
pend[, ediff := 100 * (T12 - T20)]
```
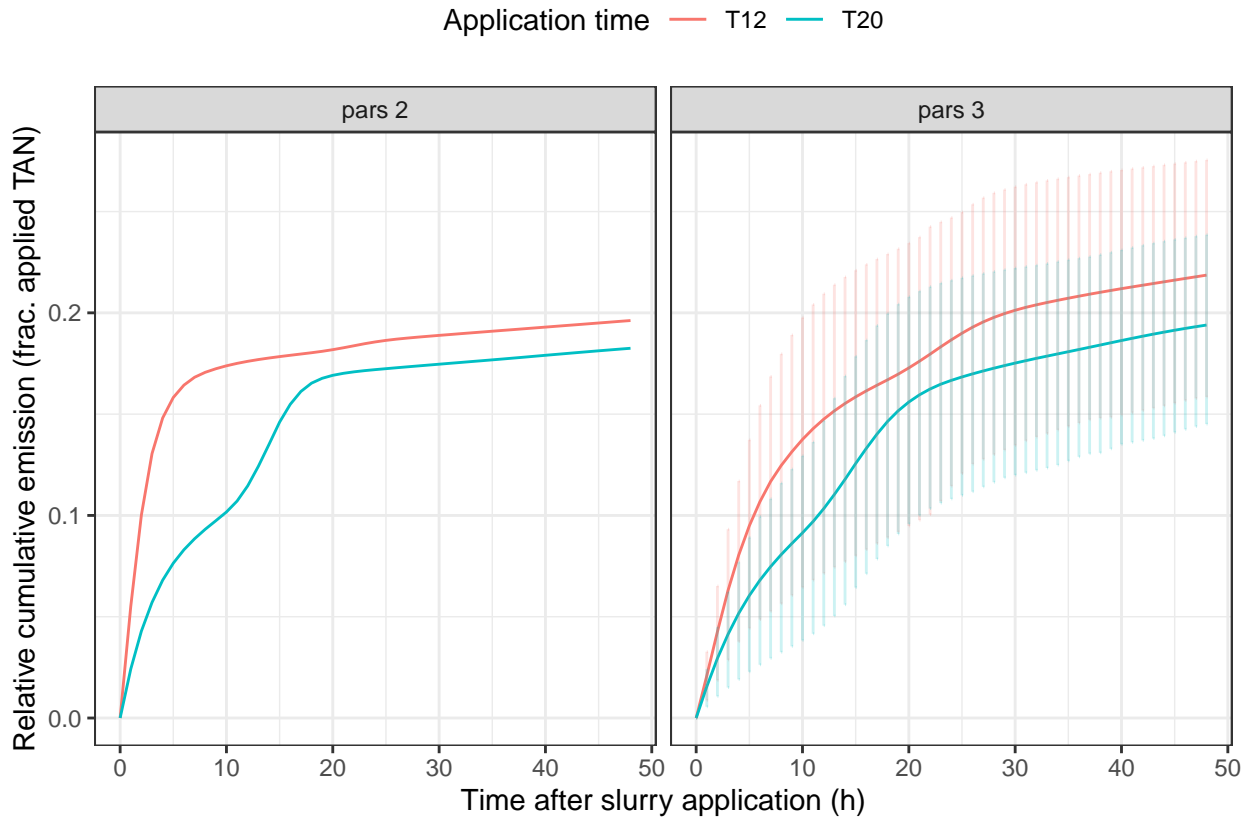
Here are the results. First cumulative emission over time.

```r
ggplot(preds, aes(ct, er, colour = apptime)) +
  geom_line() +
  geom_errorbar(aes(ymin = er.lwr, ymax = er.upr, width = 0), alpha = 0.2) +
  facet_wrap(~ parset) +
  theme_bw() +
  theme(legend.position = 'top') +
  labs(x = 'Time after slurry application (h)',
       y = 'Relative cumulative emission (frac. applied TAN)',
       colour = 'Application time')
```

Error bars are 90% confidence intervals.

Difference between 12:00 and 20:00 as percentage of applied TAN.

```
print(pend)
```

```
## Key: <parset>
##    parset       T12        T20      ediff
##    <char>     <num>      <num>      <num>
## 1: pars 2 0.1962045 0.1825097 1.369486
## 2: pars 3 0.2186139 0.1939791 2.463477
```

3. Confidence interval on comparison with parameter set 3

Run ALFAM2 model, returning all 100 sets of results (100 "plausible parameter sets", see paper), and get final times.

```
pred3c <- alfam2(din, pars = alfam2pars03, group = 'apptime', conf.int = 'all')
```

```
## Warning in alfam2(dat = dat, pars = pars, add.pars = add.pars, app.name = app.na
##      So function will return relative emission only.

## Default parameters (Set 3) are being used.

## Warning in prepDat(dat, warn = warn): Argument prep.dum = TRUE but there are no variables to convert
##    Ignoring prep.dum = TRUE.

## Warning in alfam2(dat = dat, pars = pars, add.pars = add.pars, app.name = app.name, : Incorporation
##      because argument time.incorp is NULL
##      So there is no incorporation.
##      Set check = FALSE to not drop, but then check output.
```

```
## Warning in alfam2(dat = dat, pars = pars, add.pars = add.pars, app.name = app.name, : Running with 1:
## These secondary parameters have been dropped:
##    app.mthd.os.f0
##    app.mthd.cs.f0
##    man.source.pig.f0
##    app.mthd.bc.r1
##    app.mthd.ts.r1
##    app.mthd.cs.r3
##    incorp.deep.r3
##    incorp.shallow.f4
##    incorp.deep.f4
```

```r
setDT(pred3c)
pend3c <- pred3c[ct == 48, ]
```

Reshape and get difference for each parameter set, and then confidence interval from quantiles.

```r
pw <- dcast(pend3c, par.id ~ apptime, value.var = 'er')
pw[, dd := T12 - T20]
print(100 * quantile(pw$dd, c(0.05, 0.5, 0.95)))
```

```
##       5%      50%      95%
## 1.312387 2.245726 3.548143
```

So 90% CI is [1.3, 3.5] % of applied TAN. This is relatively high precision in my opinion. In includes the uncertainty in parameter values that come from variability in measurements among institutions. It might seem strange that we have high confidence in a positive difference here in the comparison (higher emission at 12:00 than 20:00) when the error bars in the plot above overlap quite a bit. The explanation is that we have more confidence in relative effects than in absolute emission, probably mainly due to the nature of the measurement data.