# KPP Stem and Log SWV : Progress Jul 2014

JH; MC, TB - TLL; C - Calibre; ML - SWI; GS, TH, NM - Tui

## Aims

* Determine what is causing the partially poor correlation between stem velocity and average log velocity and if possible fix it.

## Things Done or In Progress

Since the [last progress update](https://rawgit.com/hammockman/scraps/master/KPP_stemVlogSWV_update_27may2014.html):

* GS working on upgrading the USNR software. Once completed this will fix the stemId problem and USNR image files will be written with the right filenames.
* GS made minor fixes to KPP PLC code relating to stem lengths and velocity logic.
* Modifications made to the SWIPC database and application (''ls15.py'') to permit real-time monitoring of stem velocity estimate based on log velocities (''avgLogSWV'')
* JH and GS onsite 24 Jul 2014.
* Checked transposition of stems on sequence deck post-sonic tool. Not a frequent occurence (though perhaps when it is occuring it is frequent, e.g. GS suggests when lugs out of alignment)
* Checked on lengths being supplied to sonic tool.
* Enabled saving of stem sonic tool raw data (.dat files).
* SWIPC database modified and processing code (''plotdat.py'',''processdat.py'') written to incorporate raw stem sonic data.
* Independent peak finding algorithm used to extract stem velocities from raw sonic data for a subset of stems.

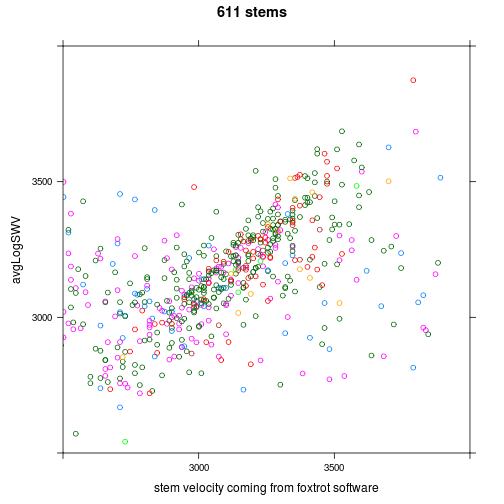
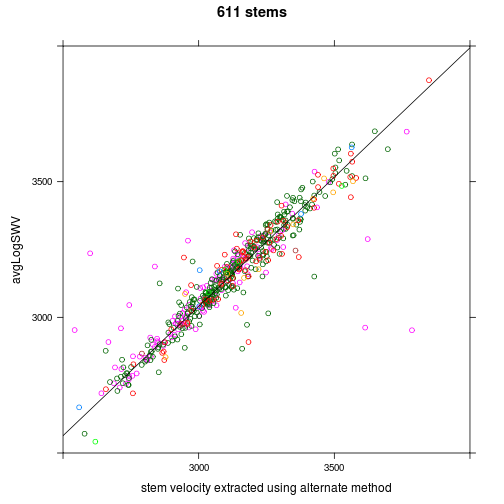
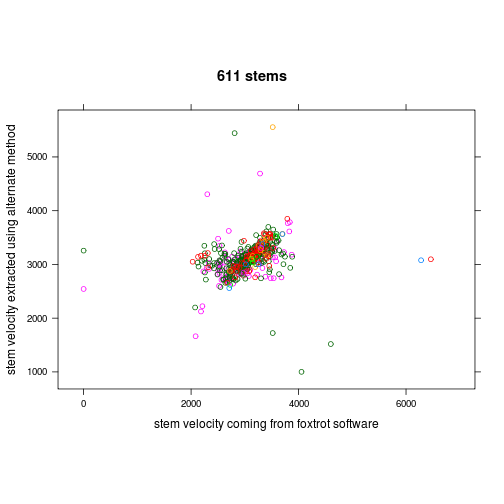
## Results

A subset of stems was identified, for which: \* raw sonic data was collected \* more than 90% of the original length was recovered as logs, \* all recovered logs have good velocities ( m/s) and lengths ( m).

An independent, alternate method was employed to extract a stem evlocity from the raw data.

The first plot below shows the usual noisy relationship between avgLogSWV and stem velocity coming from the software running on foxtrot. The second shows the very much improved relationship when the alternate velocity extraction method is used . This implies that:

1. the stem numbering and tracking system is working, at least in most cases
2. the stem-log velocity 'noise' arises in the foxtrot software, perhaps due to echo/peak finding algorithm shortcomings or perhaps due to a bug in the result labelling.

How tight is the relationship between my stem velocity and avgLogSWV?

summary(m <- lm(avgLogSWV ~ myvel, M, subset = abs(avgLogSWV - myvel) < 150))

##   
## Call:  
## lm(formula = avgLogSWV ~ myvel, data = M, subset = abs(avgLogSWV -   
## myvel) < 150)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -175.41 -25.50 3.32 30.08 108.07   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.16e+02 2.87e+01 7.54 2.2e-13 \*\*\*  
## myvel 9.43e-01 9.17e-03 102.82 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 43.2 on 508 degrees of freedom  
## (67 observations deleted due to missingness)  
## Multiple R-squared: 0.954, Adjusted R-squared: 0.954   
## F-statistic: 1.06e+04 on 1 and 508 DF, p-value: <2e-16

# hist(m$residuals, 13, xlim=c(-1,1)\*100) qqnorm(m$residuals)  
# mean(M$avgLogSWV-M$myvel, na.rm=TRUE)

Are the current confidence measures any use? The subset stems can be classed as: - good: the alternate velocity extraction method works and yields a stem velocity well correlated with the log average. - bad: the alternate velocity extraction method works, but the stem-log data point is an outlier. - unusable: no alternate velocity can be computed.

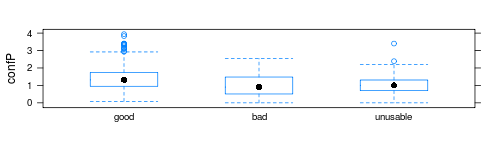
The next pair of plots indicate that the current confidence measures produced by the foxtrot software cannot distinguish between these classes.

# densityplot(~confF | signal.class, M, layout=c(3,1)) densityplot(~confP |  
# signal.class, M, layout=c(3,1))  
bwplot(confF ~ signal.class, M)



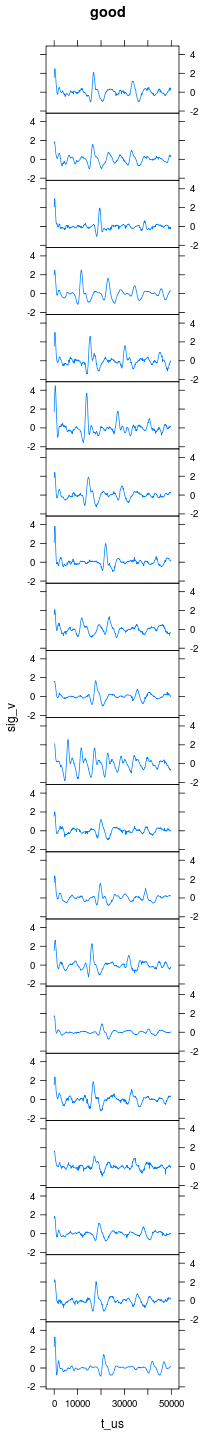
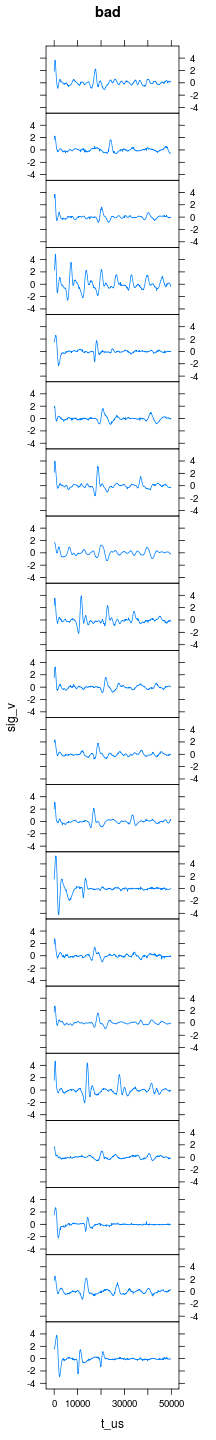
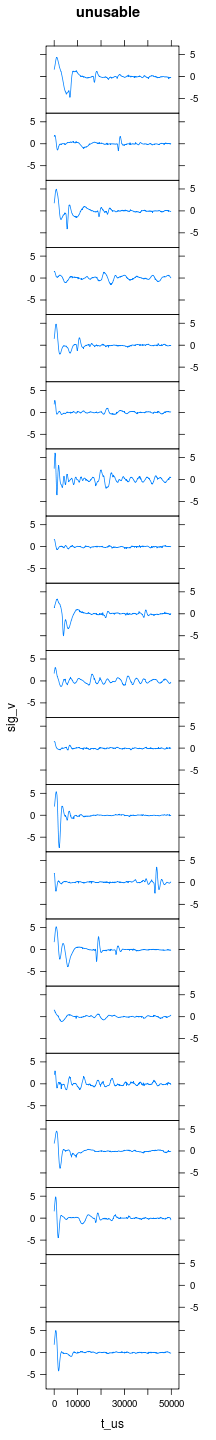
plot of chunk unnamed-chunk-3

bwplot(confP ~ signal.class, M)



plot of chunk unnamed-chunk-3

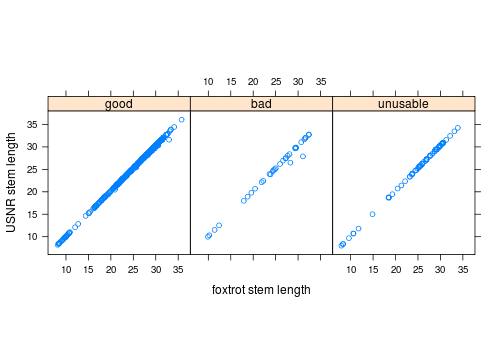
How do the raw signals vary between classes? The next plots show examples of signals from the three classes described above.

Unsurprisingly, signals in the 'unusable' class lack clear echos, and consequently do not permit a velocity to be extracted. Much less obvious is what distinguishes 'good' from 'bad', perhaps the divergence originates from other than the stem sonic raw signal (e.g. stem length, log characteristics, alternate echo extraction method)

Quickly check lengths:

xyplot(stemLength ~ length.1 | signal.class, M, layout = c(3, 1), xlab = "foxtrot stem length",   
 ylab = "USNR stem length", aspect = "iso")



plot of chunk unnamed-chunk-5

A couple of the 'bad' lengths are dodgy, but length dosn't look like a good explanation.

## Overall Stem Tool Performance

The following table details stem sonic measurement outcomes using both the current software (standard) and the alternate velocity extraction algorithm (alternate):

|  |  |  |
| --- | --- | --- |
|  | Standard % | Alternate % |
| stem not hit (i.e. 5-digit stem number) | 42 | 42 |
| no raw stem sonic data | 8 | 8 |
| raw sonic data unusable | 5 | 5 |
| extracted stem velocity differs significantly from avgLogSWV | 22 | 3 |
| good stem velocity | 23 | 42 |
| total | 100 | 100 |

## Conclusions and Recommendations

* The stem sonic tool (both hardware and software) needs replacing/re-engineering in order to:
* improve the fraction of stems measured,
* improve the extraction of velocity from the raw data.
* If the existing tool hardware is not replaced, it is very likely that better velocity estimates can be extracted from the raw data currently being collected.
* Any future system needs to come with a confidence estimate that accurately reflects the measurment quality.
* A new stem sonic tool would be better positioned on the chain after the sequence deck, rather than in the current position.
* Keep Greg sweet, he appears essential to maintaining an operational system.