

# Figure 3 and S6 changes in downstream promoter sequence alter polii pausing

Wanqing Shao(*was@stowers.org*)

## Contents

Description	1
Enviroment setup	1
Analysis	2
Pol II ChIP-nexus profile after TRI treatment . . . . .	2
Quantify Pol II changes . . . . .	6
Session Info	7

## Description

This set of experiments is aimed at testing whether reporter-ChIP-nexus can detect changes in Pol II pausing after manipulating promoter sequences. To do this, we focused on downstream promoter sequences, as accumulating evidence suggests that downstream DNA influences Pol II pausing. We took two TATA promoters *Act5C* and *pepck*, and changed its downstream sequence to that of a pausing promoter *pk* or *dve*. Changes in paused Pol II stability were probed by performing Pol II ChIP-nexus at control and Triptolide (TRI) treated conditions.

## Enviroment setup

```
library(GenomicRanges, warn.conflicts=F)
library(magrittr)
library(Rmisc)

setwd("/data/analysis_code")
options(knitr.figure_dir =
  "Figure3_S6_changes_in_downstream_promoter_sequence_alter_polii_pausing")

source("shared_code/knitr_common.r")
source("shared_code/ggplot_common.r")
source("shared_code/granges_common.r")
source("shared_code/metapeak_common.r")
source("shared_code/sample_common.r")
```

## Analysis

### Pol II ChIP-nexus profile after TRI treatment

```

plasmid_annotations <- import("./plasmid_annotation.bed")

get_exo_metapeak <- function(sample, upstream=100, downstream=101,
                             smooth=NA, endogeneous = F, dps_sample_path=NULL){

  gene <- as.character(subset(sample_list, sample_name == sample)$gene)
  chromosome <- as.character(subset(sample_list, sample_name == sample)$chromosome)

  if(endogeneous ==F){

    sample_path <- load_bigwig(sample)
    region <- plasmid_annotations[seqnames(plasmid_annotations) == chromosome &
                                plasmid_annotations$name == gene] %>%
      resize(., 1, "start")
    seqlevels(region) <- chromosome
    metapeak <- exo_metapeak(region, sample_path,
                            upstream=upstream, downstream=downstream,
                            sample_name=gene, smooth=smooth)
    metapeak$sample <- paste(metapeak$sample_name, metapeak$strand)
    metapeak

  }else{

    region <- genome_annotations[grep(gene, genome_annotations$name, ignore.case = T)]
    seqlevels(region) <- as.character(seqnames(region))
    metapeak <- exo_metapeak(region, dps_sample_path,
                            upstream=upstream, downstream=downstream,
                            sample_name=gene, smooth=smooth)
    metapeak$sample <- paste(metapeak$sample_name, metapeak$strand)
    metapeak

  }
}

compare_dms0_and_tri <- function(dms0, tri, name, plotting = T){

  plasmid_dms0 <- get_exo_metapeak(dms0, upstream=150, downstream = 151)
  plasmid_tri <- get_exo_metapeak(tri, upstream=150, downstream = 151)

  plasmid_tri$reads <- plasmid_tri$reads / sum(abs(plasmid_dms0$reads))
  plasmid_dms0$reads <- plasmid_dms0$reads / sum(abs(plasmid_dms0$reads))

  plasmid_dms0$sample_name <- paste(name, "Control")
  plasmid_tri$sample_name <- paste(name, "TRI")

  if(plotting){
    metapeak <- rbind(plasmid_dms0, plasmid_tri)
    metapeak.p <- subset(metapeak, strand == "+")
  }
}

```

```

metapeak.n <- subset(metapeak, strand == "-")

plot <- ggplot(metapeak.p, aes(x = tss_distance, y = reads, fill = strand))+
  geom_bar(fill="#B23F49", stat="identity") +
  geom_vline(xintercept =0, linetype = "dotdash")+
  geom_bar(data=metapeak.n, aes(x=tss_distance, y=reads),
           fill="#045CA8", stat="identity")+
  xlab("Distance from TSS (bp)") + ylab("Normalized reads") +
  facet_wrap(facets = "sample_name", ncol =2 )
print(plot)
}

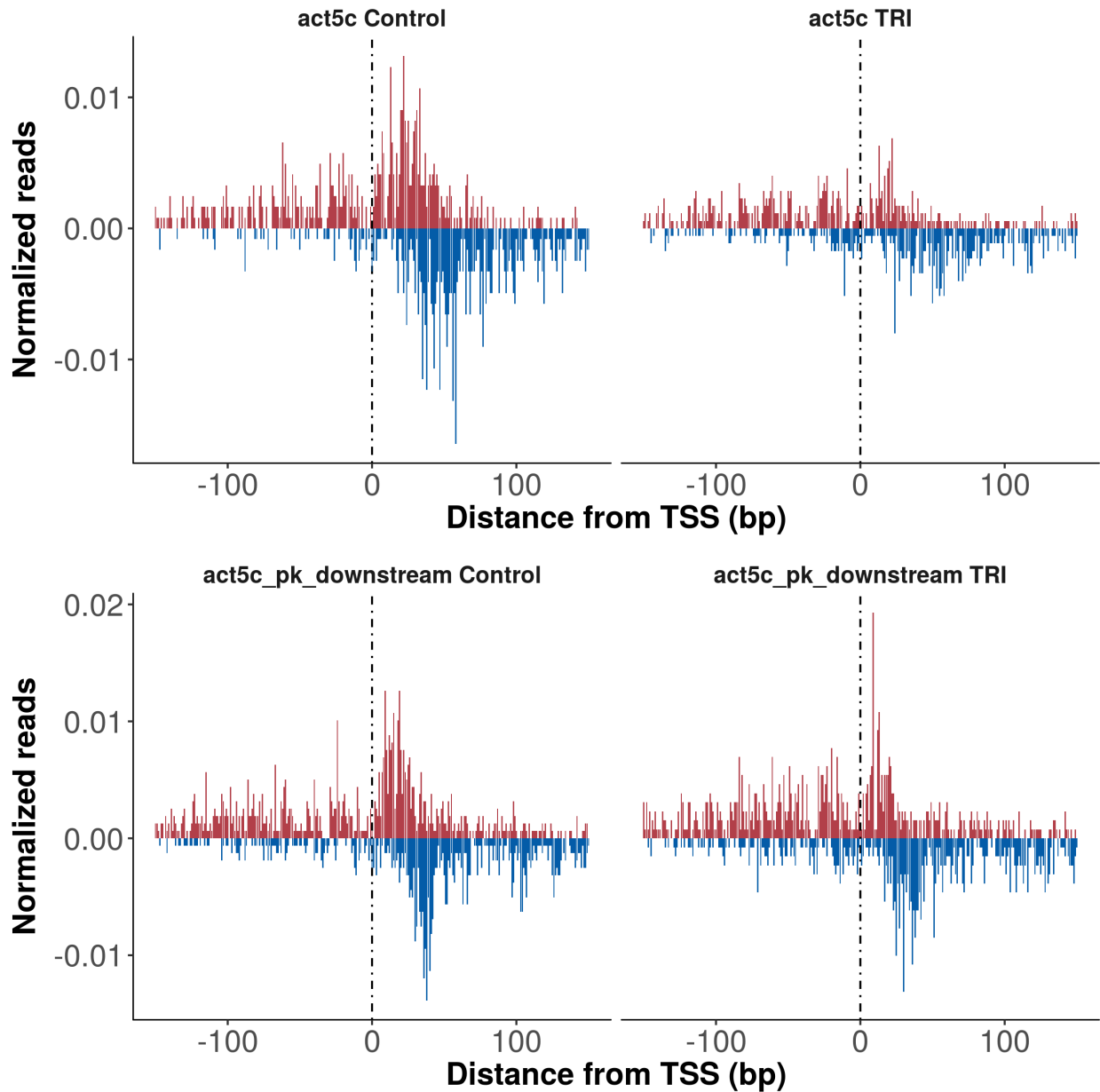
dmsig <- subset(plasmid_dms, tss_distance >0 & tss_distance <= 80)$reads %>%
  abs() %>% sum()
tri_sig <- subset(plasmid_tri, tss_distance >0 & tss_distance <= 80)$reads %>%
  abs() %>% sum()

sig_df <- data.frame(condition = c("dms", "tri"),
                    paused_polii = c(dmsig, tri_sig),
                    name = name)
sig_df$paused_pol_norm <- sig_df$paused_polii / sig_df$paused_polii[1]
sig_df
}

name_list <-c("act5c", "act5c_pk_downstream")

act5c_pol_sig_rep1 <- mapply(compare_dms_and_tri,
  paste0("reporter_dms_5m_dps_", name_list, "_rpb3_chipnexus_rep1"),
  paste0("reporter_tripolide_5m_dps_", name_list, "_rpb3_chipnexus_rep1"),
  name_list, SIMPLIFY = F, USE.NAMES =F) %>% do.call(rbind, .)

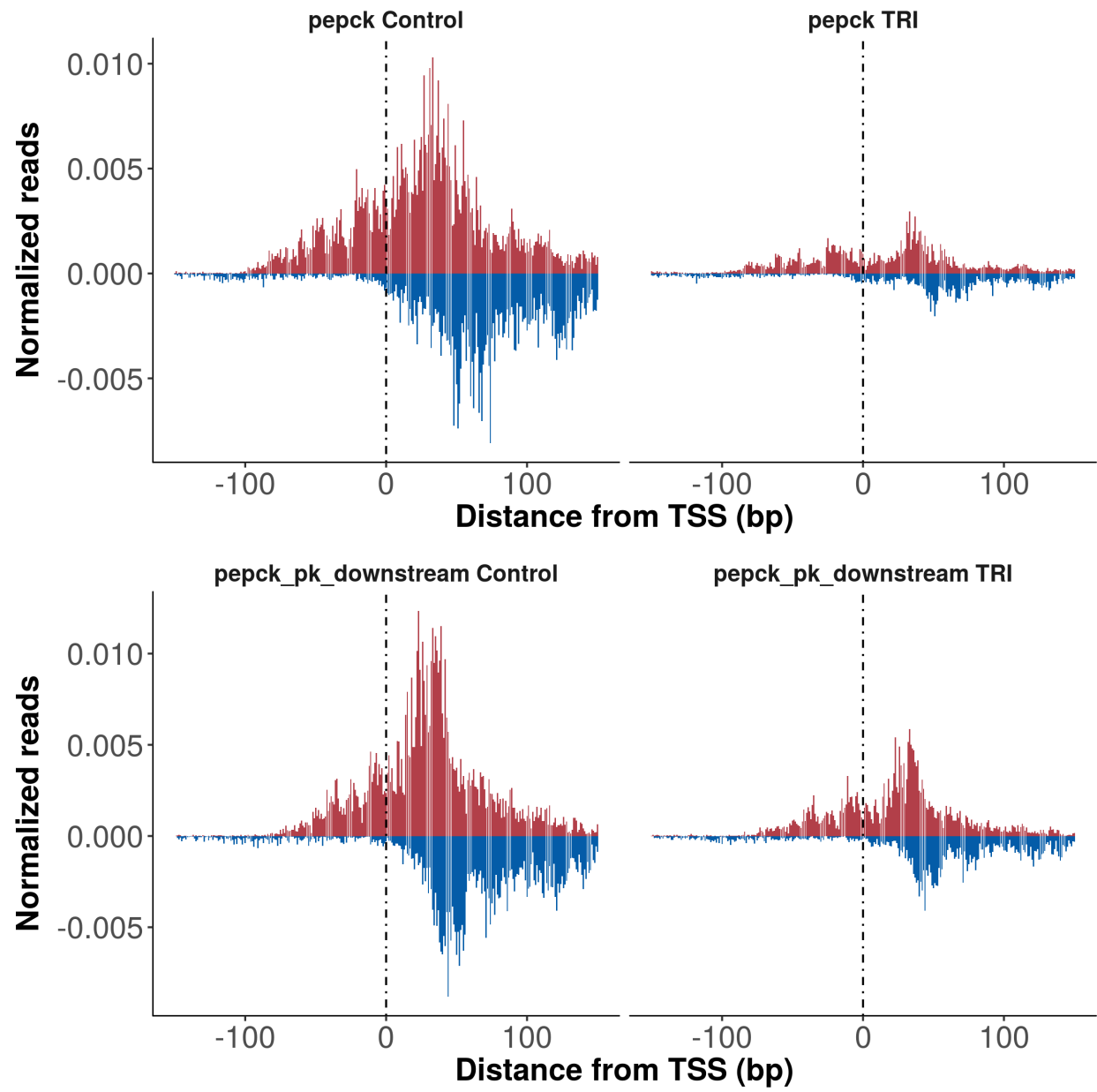
```

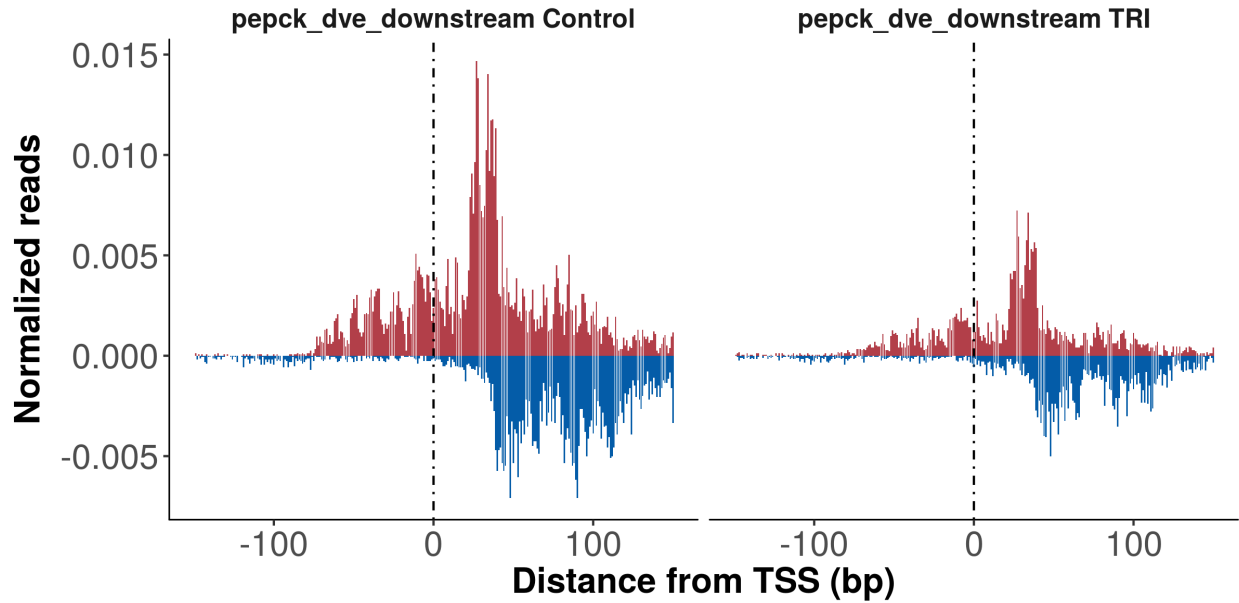


```
act5c_pol_sig_rep2 <- mapply(compare_dms0_and_tri,
  paste0("reporter_dms0_5m_dps_", name_list, "_rpb3_chipnexus_rep2"),
  paste0("reporter_triaptolide_5m_dps_", name_list, "_rpb3_chipnexus_rep2"),
  name_list, list(F), SIMPLIFY = F, USE.NAMES = F) %>% do.call(rbind, .)

name_list2 <-c("pepck", "pepck_pk_downstream", "pepck_dve_downstream")

pepck_pol_sig_rep1 <- mapply(compare_dms0_and_tri,
  paste0("reporter_dms0_5m_dps_", name_list2, "_rpb3_chipnexus_rep1"),
  paste0("reporter_triaptolide_5m_dps_", name_list2, "_rpb3_chipnexus_rep1"),
  name_list2, SIMPLIFY = F, USE.NAMES = F) %>% do.call(rbind, .)
```





```
pepck_pol_sig_rep2 <- mapply(compare_dms0_and_tri,
  paste0("reporter_dms0_5m_dps_", name_list2, "_rpb3_chipnexus_rep2"),
  paste0("reporter_tripitolide_5m_dps_", name_list2, "_rpb3_chipnexus_rep2"),
  name_list2, list(F), SIMPLIFY = F, USE.NAMES = F) %>% do.call(rbind, .)
```

## Quantify Pol II changes

```
process_pol_sig <- function(df, control_n = 1){
  df_sub <- subset(df, condition == "tri")
  df_sub$paused_pol_norm <- df_sub$paused_pol_norm / df_sub$paused_pol_norm[control_n]
  df_sub
}

sig_df_list <- list(act5c_pol_sig_rep1, act5c_pol_sig_rep2,
  pepck_pol_sig_rep1, pepck_pol_sig_rep2)

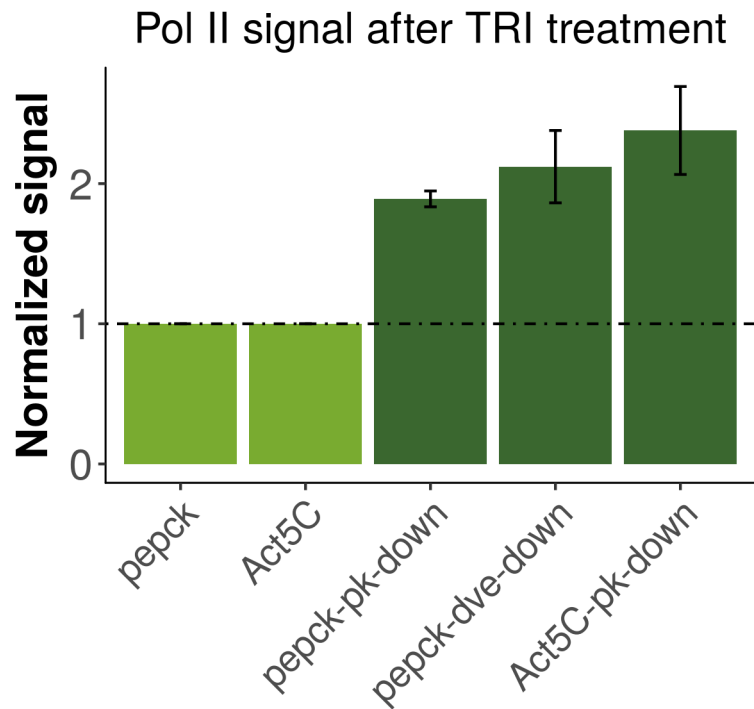
sig_df <- lapply(sig_df_list, process_pol_sig) %>% do.call(rbind, .)

summary_df <- summarySE(sig_df, measurevar="paused_pol_norm",
  groupvars=c("name", "condition"))

summary_df$name <- factor(summary_df$name,
  levels = c( "pepck", "act5c",
    "pepck_pk_downstream",
    "pepck_dve_downstream",
    "act5c_pk_downstream"))

ggplot(summary_df, aes(x=name, y=paused_pol_norm)) +
  geom_bar(stat= "identity", position = "dodge",
    fill = c(rep("#79AB30", 2), rep("#3A672F", 3))) +
  geom_errorbar(aes(ymin=paused_pol_norm-se, ymax=paused_pol_norm+se),
    width=.1, position=position_dodge(.9)) +
```

```
ggtitle("Pol II signal after TRI treatment")+
ylab("Normalized signal")+
xlab("")+
scale_x_discrete(labels=c("pepck", "Act5C", "pepck-pk-down",
                          "pepck-dve-down", "Act5C-pk-down")) +
geom_hline(yintercept = 1, lty = 4) +
theme(axis.text.x = element_text(size=14, angle = 45, hjust = 1))
```



## Session Info

This analysis was performed with the following R/Bioconductor session:

```
sessionInfo()
```

```
## R version 3.4.4 (2018-03-15)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.4 LTS
##
## Matrix products: default
## BLAS: /usr/lib/libblas/libblas.so.3.6.0
## LAPACK: /usr/lib/lapack/liblapack.so.3.6.0
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8      LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8      LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8  LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8     LC_NAME=C
##  [9] LC_ADDRESS=C             LC_TELEPHONE=C
```

```

## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] parallel stats4      stats      graphics  grDevices  utils      datasets
## [8] methods    base
##
## other attached packages:
## [1] reshape2_1.4.3      rtracklayer_1.38.3  ggplot2_2.2.1
## [4] pander_0.6.1        Rmisc_1.5           plyr_1.8.4
## [7] lattice_0.20-35     magrittr_1.5        GenomicRanges_1.30.3
## [10] GenomeInfoDb_1.14.0 IRanges_2.12.0      S4Vectors_0.16.0
## [13] BiocGenerics_0.24.0
##
## loaded via a namespace (and not attached):
## [1] Rcpp_0.12.17          compiler_3.4.4
## [3] pillar_1.2.3          XVector_0.18.0
## [5] bitops_1.0-6          tools_3.4.4
## [7] zlibbioc_1.24.0       digest_0.6.15
## [9] evaluate_0.10.1       tibble_1.4.2
## [11] gtable_0.2.0          rlang_0.2.1
## [13] Matrix_1.2-14         DelayedArray_0.4.1
## [15] yaml_2.1.19           GenomeInfoDbData_1.0.0
## [17] stringr_1.3.1         knitr_1.20
## [19] Biostrings_2.46.0     rprojroot_1.3-2
## [21] grid_3.4.4            Biobase_2.38.0
## [23] XML_3.98-1.11         BiocParallel_1.12.0
## [25] rmarkdown_1.10        matrixStats_0.53.1
## [27] GenomicAlignments_1.14.2 backports_1.1.2
## [29] scales_0.5.0          Rsamtools_1.30.0
## [31] htmltools_0.3.6       SummarizedExperiment_1.8.1
## [33] colorspace_1.3-2      labeling_0.3
## [35] stringi_1.2.3         RCurl_1.95-4.10
## [37] lazyeval_0.2.1        munsell_0.5.0

```