NCT02578901

A-TREAT Statistical and Analytical Plan

2019-06-03

9.7 Statistical Methods

9.7.1 Statistical and Analytical Plan

All statistical inference will be based on a two-sided 0.05 level of significance and two-sided 95% confidence intervals.

Study Populations

Subjects will be grouped according to randomization groups: TXA versus placebo. Unless otherwise specified, all efficacy analyses will be by a modified intention to treat (mITT) strategy among activated subjects as described below. All subjects who were activated to study drug will be included in the analyses in the treatment group to which they were randomly assigned. Subjects who were activated for study drug will be included in the analyses, even if they stop study drug "early", cross over between treatment groups, or receive prophylactic transfusions not in accordance with the protocol.

1. Efficacy Population

The mITT population used for efficacy analyses will be comprised of all randomized patients for whom an order for administration of study drug is activated. Efficacy analyses on these patients will use data gathered from the time the order for study drug was activated until 30 days post activation. Randomized patients who develop exclusion criteria prior to activation of study drug and who thus are excluded from receipt of study drug and patients whose platelet counts never drop below the 30K threshold for activation of study drug will not be included in the primary efficacy population.

2. Safety Population

The population used for safety analyses will be all patients who receive any amount of study drug. Follow-up for mortality and thrombotic events will occur at 120 days post activation. Other adverse events and serious adverse events will be collected for 30 days post discontinuation of study drug, with visual examinations occurring weekly for two weeks after discontinuation of study drug. SOS (VOD), AEs, and SAEs will be based on clinical diagnosis and patient report during the surveillance period. The frequency of follow-up will differ between inpatients and outpatients, with the former based on daily visits by the research coordinators and the latter based on patient diaries and semiweekly clinic visits (while still on study drug) or weekly contacts (after discontinuation of study drug

Demographic and Baseline Characteristics

Demographic and baseline characteristics will be presented in tables stratified by treatment arm, clinical site and therapeutic group (allogeneic transplant, autologous transplant, or leukemia). Demographic variables will include age, sex, race, ethnicity, height, weight, BMI and primary diagnosis. Baseline lab values will include platelets, hematocrit, serum creatinine, prothrombin time (PT), international normalized ratio (INR), partial thromboplastin time (PTT), fibrinogen, D-dimer, thrombin time, hemoglobin and ~~HLA PRA~~. Subject dispositions will also be presented, and the table will indicate total screened, consented, randomized, activated, completed therapy, safety follow-up and long-term follow-up.

Primary Efficacy Analyses

The odds of bleeding at WHO grade 2 level or above will be analyzed in a logistic regression model adjusting for treatment arm, clinical site as a factored variable, and therapeutic group (allogeneic transplant, autologous transplant, or leukemia) as a factored variable. The test statistic will be a Wald test based on the score for the

treatment arm parameter from the logistic regression model with adjustment for multiply imputed missing data.

$$\log\frac{p}{1-p} = \beta_0 + \beta_1 X_t + \beta_2 Z_1 + \beta_3 Z_2 + \beta_4 W_1 + \beta_5 W_2$$

In the above full model, p refers to the probability of a WHO grade 2+ bleed conditional on the covariates. β_0 represents the log odds of bleeding for a subject in the placebo group, and in the reference site and reference therapeutic group (UNC and allogeneic, respectively). β_1 is the log odds ratio of bleeding between placebo and treatment arms for fixed site and therapeutic group. The parameters β_2, β_3 represent the log odds ratio between UPT and UNC and between UWM and UNC, respectively, for a fixed treatment arm and therapeutic group. Similarly, β_4, β_5 are the log odds ratios between autogenic and allogenic and between chemotherapy and allogenic groups, for a fixed site and treatment arm. Z_1, Z_2, W_1, W_2 are indicator variables: $Z_1 = 1$ if the subject is from UPT, $Z_2 = 1$ if from UWM, $W_1 = 1$ if in autogenic therapy, $W_2 = 1$ if in chemotherapy.

The analysis of the primary endpoint, in order to adjust for multiply imputed data, will use the score for the treatment parameter evaluated at the null hypothesis in a Wald test. In this case, the null hypothesis θ_0 has the treatment parameter $\beta_1 = 0$ and all other $\beta_j = \hat{\beta}_j$ $(j = 0,2,3,4,5)$ their respective maximum likelihood estimates. We give the form of the relevant score below (where X is the matrix of covariates, X_i refers to the ith row and X_{1i} refers to the element in the ith row and the column corresponding to β_1).

$$U_1 = \sum_{i=1}^{n} X_{1i}(Y_i - p_i(\theta_0))$$

$$p_i(\theta_0) = \frac{e^{X_i\theta_0}}{1 + e^{X_i\theta_0}}$$

The variance of the score is the element of the observed Information matrix (derivative of the score) that corresponds to the score of interest.

$$Var(U_1) = \sum_{i=1}^{n} X_{1i}^2 p_i(\theta_0)\big(1 - p_i(\theta_0)\big)$$

Further discussion is presented in the missing data section along with R code for performing the analysis.

Secondary Analyses

The secondary endpoints of mean number of platelet transfusions and the mean number of days alive without WHO grade 2 level or above bleeding will each be analyzed in a linear regression model adjusting for treatment arm, clinical site as a factored variable, and therapeutic group (allogeneic transplant, autologous transplant, or leukemia) as a factored variable. The test statistic will be based on the Wald statistic (parameter estimate divided by its standard error) as computed for the treatment arm parameter from the linear regression model using the Huber-White sandwich estimator for standard errors to account for possible heteroscedasticity and adjusting for multiply imputed data.

```
lmTrans <- lm(ptrans ~ treatment +
              site + group)
lmWHODays <- lm(days ~ treatment +
              site + group)
```

Other Analyses

Analysis of the supportive and exploratory endpoints will be conducted using regression models adjusting for treatment arm, clinical site as a factored variable, and therapeutic group (allogeneic transplant, autologous transplant, or leukemia) as a factored variable. Linear regression will be used for endpoints measuring the days free of bleeding and highest grade of bleeding, logistic regression will be used for binary endpoints, and proportional hazards regression will be used for times to event. The Huber-White sandwich estimator of the standard error will be used to allow for departures from the model based variance estimates, and adjustment will be made for multiply imputed missing data.

```
library("survival")
lmBleedFree <- lm(free ~ treatment +
               site + group)
lmHiBleed <- lm(hibleed ~ treatment +
               site + group)
lmBleedDth <- glm(bleeddth ~ treatment +
               site + group,family="binomial")
lmFirstBleed <- coxph(fBleed ~ treatment +
               site + group)
lmPltTrans <- lm(pltTrans ~ treatment +
               site + group)
lmRedTrans <- lm(redTrans ~ treatment +
               site + group)
lmNoPlt <- glm(noPlt ~ treatment +
               site + group,family="binomial")
lmNoRed <- glm(noRed ~ treatment +
               site + group,family="binomial")
lmNadir <- glm(hiNadir ~ treatment +
               site + group,family="binomial")
```

Subgroup Analyses

WHO grade 2 or above bleeding will be assessed separately among patients who did and did not have bleeding at randomization, as well as within each of the therapeutic groups of allogeneic transplant, autologous transplant and leukemia. This will be performed similar to regression models above by restricting to the dataset as described.

Safety Analyses

The safety endpoints will be tabulated and presented without inferential statistics. These endpoints, AEs, SAEs, thrombotic events, incidence of VOD, all-cause mortality, and death due to thrombosis, will each be tabulated by treatment arm and primary diagnosis group.

Missing Data

Patients will be followed for all clinical trial outcomes regardless of their adherence to the prescribed regimen for the study drug (TXA or placebo). Very few patients are expected

to withdraw consent, though there will undoubtedly be some cases in which the data are missing for one or more of the primary, secondary, or supporting analyses due to withdrawal of consent or competing risks.

Initial analyses of primary and secondary endpoints will be based on missing-at-random (MAR) models that condition on the patients' absolute neutrophil counts (ANC), site, therapeutic group and days since randomization to study drug. In the MAR model, data will be imputed by assuming that patients with missing data will be presumed to have distributions of outcomes the same as that observed in comparable patients who have complete data.

Sensitivity analyses for the primary endpoint will use missing not at random (MNAR) models to reflect increased or decreased odds of bleeding among patients with incomplete data. Each treatment arm will be parameterized separately, and the robustness of any estimated treatment effect to those varying levels of dependence between early death or withdrawal of consent and incidence of bleeding will be quantified. Additional models will consider separate mechanisms for early death and withdrawal of consent.

Primary Endpoint
The model for the primary endpoint will be proportional hazards of the effect of treatment arm stratified by therapeutic group. The model will include a time-varying binary covariate of high (greater than 500) versus low (less than 500) ANC and a factored site variable.

The following model will be fit for complete-case data to estimate the baseline hazard for each treatment stratum. Then the residual hazard will be estimated for censored subjects based on observed subjects within the same site and with the same ANC state on the day of censoring. Sampling from this hazard function will provide imputed values for use in the primary analysis.

$$\lambda_Z(t) = \lambda_{Z_0}(t) \exp\left\{\beta_{1,Z} A_i(t) + \sum_{j=1}^{3} \beta_{j+1,Z} 1_{C_i=j}\right\}$$

λ_Z represents the hazard function for the Zth treatment arm, λ_0 the baseline hazard, $A_i(t)$ the most recent ANC state on day t for the ith subject and $1_{C_i=j}$ being the indicator function for whether the subject belonged to the jth site.

$$\lambda^{(l)}(t) = \lambda_{Z_0}(t) \exp\left\{\beta_{1,Z} A^{(l)}(t) + \sum_{j=1}^{3} \beta_{j+1,Z} 1_{c^{(l)}=j}\right\}$$

This model represents the residual hazard for the lth subject with missing data; $A^{(l)}(t)$ is the most recent ANC state on day frozen at the day of censoring for the lth subject (described further in implementation below); the other parameters remain the same as before.

During sensitivity analyses, to reflect the MNAR models, the first model above will be scaled with a sensitivity term α_Z for each treatment arm to represent increased or decreased odds of bleeding, so we will sample from the hazard $\alpha_Z \lambda_Z(t)$ for a given treatment arm. A tipping point analysis will be conducted by performing the imputation and primary analysis using the following values for each α_Z: 0.75, 0.80, 0.85, 0.90, 0.95,

1.00, 1.05, 1.10, 1.15, 1.20, 1.25. Determining how far the analysis can deviate from the MAR model with significant results will contextualize and improve the validity of the conclusions of the study.

The process for coding this in R follows. First, the data are collected into a “wide” data frame, which contains columns for case ID, case site, therapeutic group, date of activation, number of days on study, indicator of WHO grade 2+ bleed (1 if a bleed occurred within 30 days of activation, 0 if no bleed occurred but the subject was followed for 30 days of activation, or NA if withdrawn for any reason before 30 days without a bleed), date of first bleed (NA if no bleed) and then a variable for the ANC count (above or below 500) for each of the 30 days post-activation. For the purposes of this analysis, the day of activation is considered to begin at midnight of the first day, bleeding occurred at noon on the day of recording and the ANC count recorded is the first measured of the day.

At the same time, missing ANC counts are filled in with the most recent ANC count (i.e. if a subject’s ANC count is missing, then it is replaced with the last non-missing ANC count that the subject had).

Then this dataset is changed into a “long” format, where each subject has as many rows as the smallest of 30, days until first bleed, and days until withdrawn from study. Keeping the other variables constant across the rows, the 30 ANC count variables are collapsed into a single variable corresponding to the daily ANC count (again, above or below 500).

This long dataset, restricted to the treatment arm and only for rows with non-missing values for bleed, becomes the argument for the “coxph” function in R, which is used to estimate the baseline hazards and regression coefficients for treatment arm.

Then a single imputation step is as follows after specifying the seed 20170615 (first day of screening). Each subject with a missing value for bleed is to be filled in. First, the “wide” dataset is trimmed at the day of censoring for such a subject by removing all ANC counts corresponding to days after withdrawal (NA values). The last remaining column is then copied to re-fill the removed columns until 30 days of ANC counts exist. This effectively “freezes” the ANC counts to the day of censoring. This “wide” dataset is once again converted to long, restricted to the particular subject missing bleed data. The new long dataset, coupled with the previously estimated baseline hazard corresponding to the appropriate treatment arm, is used to estimate a residual hazard function unique to the subject. This is the function from which a single sample is drawn; if the sample corresponds to a bleed within the 30 day period, then the previously-missing bleed is filled as 1, otherwise it is 0. Then this process is repeated for all rows with missing values for bleed. Finally, the primary efficacy analysis is conducted and the score for the treatment variable and its variance recorded.

Then the imputation is repeated $k = 20$ times to produce 20 different values for the score and variance in the primary efficacy analysis. These are combined into an estimated score, which is the mean of the imputed scores and an estimated variance, which is the sum of the mean of the imputed variances and the variance of the imputed scores. Then a Wald test will compare the combined estimate and standard error with the standard normal distribution.

$$p = 2 * Pr\left(\left|\frac{\widehat{\theta_k}}{\widehat{SE}(\widehat{\theta_k})}\right| > z_\alpha\right)$$

The code is attached in an appendix.

Secondary Endpoints
We will use a random regression model of multiple imputation for the both of the secondary endpoints: number of days alive without WHO grade 2+ bleeding and number of platelet transfusions within 30 days after activation. In a previous version of the SAP, the imputation scheme for the platelet transfusions outcome involved hot deck imputation. This has been changed in this updated SAP to parallel the imputation scheme for the number of days alive outcome variable.

The number of days alive without such bleeding will be regressed against treatment arm, therapeutic stratum, site and number of days on study within the efficacy window. In addition, anticipating that subjects will likely either bleed numerous times or bleed relatively infrequently, an additional covariate will be included that indicates if a subject had 2 or more days in sequence of WHO grade 2+ bleeding within the efficacy window.

The model will be similar for the number of platelet transfusions within 30 days after activation: The number of such transfusions will be regressed against treatment arm, therapeutic stratum, site and number of days on study within the efficacy window. In addition, anticipating that subjects will likely either be transfused numerous times (but likely not as frequently on adjacent days) or relatively infrequently, an additional covariate will be included that indicates if a subject had 2 or more transfusions within a 7-day within the efficacy window.

First, these models will be fit for the complete-case data. The standard deviation of the error will be estimated by resampling 1000 times from the errors (difference between observed and predicted days of bleeding). Then a single imputation will fill in the outcome variable. These imputed data will be drawn from a normal distribution with mean given by the predicted values using the complete-case fit and standard deviation given by the resampled estimate from the complete-case model. With this imputed data set, the estimate for the treatment arm parameter and its standard error will be computed and recorded based on the model for the secondary endpoint analysis. The seed for imputations will be the same as above 20170615.

The imputation is then repeated $k = 20$ times to produce 20 different estimates of the parameter value and estimates of the variance. The parameter estimates will be combined by taking the mean of all of the estimates, and the standard errors will be combined by taking the square root of the sum of the mean of the k estimated variances and the variance of the k parameter estimates. Then a Wald test will compare the combined estimate and standard error with the standard normal distribution.

$$p = 2 * Pr\left(\left|\frac{\widehat{\theta_k}}{\widehat{SE}(\widehat{\theta_k})}\right| > z_\alpha\right)$$

```
set.seed(20170615)
library(sandwich)
reps <- 20 # 20 replications for the imputations below
```

```
##### days: number of days alive without WHO grade 2+ bleeding


dat <-
actBase[,c("caseid","trt","site","group","daysonstudy")]

dat$days <- dat$availDays <- dat$bleedinarow <-
rep(NA,nrow(actBase))

for (i in 1:length(actBleeds)){

  tmpBld <- actBleeds[[i]]

  availDts <- floor(tmpBld$JDates)-actBase$activ[i]+1 # all
days with any bleeding entry

  maxDate <- ifelse(actBase$stopReason[i]=="Death" &
actBase$daysonstudy[i] < 30,actBase$daysonstudy[i],30)

  effDays <- 1:maxDate


  anyMiss <- any(!effDays %in% availDts)


  bleedIdx <- as.numeric(tmpBld$maxBldgrade) >= 2 & (availDts
>= 1 & availDts <= maxDate)


  dat$bleedinarow[i] <-
any(diff(floor(tmpBld$JDates[bleedIdx])) == 1) # if any
bleeding days are adjacent

  dat$availDays[i] <-
sum(as.numeric(tmpBld$maxBldgrade[availDts >= 1 & availDts <=
maxDate]) < 2) # even if there are some days with missing
bleeds, record the available days alive without WHO 2+


  if (anyMiss){

    dat$days[i] <- NA # there is missing bleeding

  } else {
```

```
    dat$days[i] <- sum(as.numeric(tmpBld$maxBldgrade[availDts
>= 1 & availDts <= maxDate]) < 2) # "official" days without
WHO 2+ in efficacy period

  }


}


dat$predDays <- dat$predSD <- rep(NA,nrow(dat))

for (arm in 1:2){

  armTmp <- c("TXA","PBO")[arm]


  dat.arm <- dat[dat$trt==armTmp,]

  lm.complete <- lm(days ~ site + group + bleedinarow +
daysonstudy,data=dat.arm) # complete case regression


  dat$predDays[dat$trt==armTmp] <-
predict(lm.complete,dat.arm) # get the predicted values for
all in the treatment arm


  lm.errors <- residuals(lm.complete) # gets the residuals

  lm.sd <- rep(NA,1000)

  for (i in 1:1000){

    samp <- sample(lm.errors,length(lm.errors),replace=TRUE)

    lm.sd[i] <- sd(samp)

  }


  dat$predSD[dat$trt==armTmp] <- mean(lm.sd) # mean resampled
error

}
```

```
days.imps <- data.frame(est=rep(NA,reps),estVar=rep(NA,reps))
# initialize regression imputation dataset

for (i in 1:reps){

  days.predicted <- rnorm(nrow(dat),dat$predDays,dat$predSD)

  impDays <- impute(dat$days,days.predicted) # only missing
data are imputed


  # impDays <- pmax(impute(dat$days,days.predicted), # only
missing data are imputed

  #                 dat$availDays) # a predicted value cannot
be smaller than the observed value


  lmWHODays.impute <- lm(impDays ~ dat$trt +

                           dat$site + dat$group) # impDays is
outcome but covariates from analysis dataset

  days.imps[i,] <- c(coef(lmWHODays.impute)[2],

                     sandwich(lmWHODays.impute)[2,2])

}

days.est <- mean(days.imps[,1])

days.se <- sqrt(var(days.imps[,1])+mean(days.imps[,2]))


days.stat <- days.est/days.se

days.p <- 2*pnorm(abs(days.stat),lower.tail=FALSE) # wald p-
value from imputation

days.ci <- days.est+qnorm(c(0.025,0.975))*days.se


##### transf: number of platelet transfusions within first 30
days post activation
```

```
actTransf <- transfusions[names(transfusions) %in%
actBase$caseid]

ntransf <- NULL

transfrow <- NULL

for (i in 1:length(actTransf)){

  tf <- actTransf[[i]]

  dts <- tf$xPltJDates- actBase$activ[i] # changed from using
ceiling; equivalent code, but more in line with other coding

  ntransf <- c(ntransf,sum(dts >= 0 & dts < 30))

  transfrow <- c(transfrow,any(diff(dts[dts >= 0 & dts < 30])
<= 7))

}



dat <-
as.data.frame(cbind(actBase$caseid,actBase$trt,actBase[,c(2,3,
5)],ntransf,transfrow)) # only keep relevant columns

colnames(dat) <-
c("caseid","trt",colnames(actBase[,c(2,3,5)]),"transf","transf
row")



dat$predTransf <- dat$predSD <- rep(NA,nrow(dat))

for (arm in 1:2){

  armTmp <- c("TXA","PBO")[arm]



  dat.arm <- dat[dat$trt==armTmp,]

  lm.complete <- lm(transf ~ site + group + transfrow +
daysonstudy,data=dat.arm) # complete case regression



  dat$predTransf[dat$trt==armTmp] <-
predict(lm.complete,dat.arm) # get the predicted values for
all in the treatment arm
```

```
  lm.errors <- residuals(lm.complete) # gets the residuals
  lm.sd <- rep(NA,1000)
  for (i in 1:1000){
    samp <- sample(lm.errors,length(lm.errors),replace=TRUE)
    lm.sd[i] <- sd(samp)
  }

  dat$predSD[dat$trt==armTmp] <- mean(lm.sd) # mean resampled
error
}

transf.imps <-
data.frame(est=rep(NA,reps),estVar=rep(NA,reps)) # initialize
regression imputation dataset
for (i in 1:reps){
  transf.predicted <-
rnorm(nrow(dat),dat$predTransf,dat$predSD)
  imptransf <- impute(dat$transf,transf.predicted) # only
missing data are imputed
  lmWHOtransf.impute <- lm(imptransf ~ dat$trt +
                           dat$site + dat$group) # imptransf
is outcome but covariates from analysis dataset
  transf.imps[i,] <- c(coef(lmWHOtransf.impute)[2],
                       sandwich(lmWHOtransf.impute)[2,2])
}
transf.est <- mean(transf.imps[,1])
transf.se <- sqrt(var(transf.imps[,1])+mean(transf.imps[,2]))
```

```
transf.stat <- transf.est/transf.se

transf.p <- 2*pnorm(abs(transf.stat),lower.tail=FALSE) # wald
p-value from imputation

transf.ci <- transf.est+qnorm(c(0.025,0.975))*transf.se
```

9.7.2 Sample Size and Precision of Statistical Inference

1. Minimal Clinically Important Difference
   Based on results observed in the PLADO study for patients who meet the general eligibility criteria for the A-TREAT study, it is anticipated that 57% of eligible patients would experience WHO Grade 2 bleeding or higher in the absence of antifibrinolytic therapy. In such a background setting of bleeding, the study investigators anticipate that less than a 10% relative reduction in bleeding rates would not be sufficient to substantially change clinical practice, because the absolute risk reduction of 5.7% would mean that it would be necessary to treat approximately 17.5 patients in order for the treatment to have impact on 1 patient ("Number Needed to Treat" (NNT) = 17.5). As much as a 25% relative reduction in bleeding rates would likely be judged sufficient to change clinical practice, because with an absolute risk reduction of 14.25%, the NNT of approximately 7 patients might be acceptable, provided no new safety issues related to antifibrinolysis in the thrombocytopenic population are uncovered. The A-TREAT investigators hypothesize that TXA will be associated with a relative reduction of 30% or higher (absolute reduction of 17%) based on observational data relative to their prior clinical experience.
2. Sequential Stopping Rules
   The conduct of the A-TREAT study will be overseen by an independent Data and Safety Monitoring Board (DSMB) who will enhance patient safety by monitoring study progress and integrity, incidence of AEs and SAEs, and interim estimates of treatment effect on bleeding. The DSMB will be guided by a group sequential stopping rule to judge the scientific and statistical credibility of interim results on the bleeding endpoints.

   While the exact stopping rule will be chosen in discussion with the DSMB (and documented in the Statistical Analysis Plan finalized prior to the first interim analysis at which the DSMB will see unblinded data), the A-TREAT investigators propose a stopping rule that would allow early stopping only if a one-sided level 0.20 O'Brien-Fleming stopping boundary suggested that TXA was associated with more bleeding than placebo. There would be no early stopping boundary for efficacy of TXA over placebo, with the final critical value for efficacy providing a one-sided 0.025 level of significance.

   The following table presents example stopping boundaries that might correspond to an observed combined event rate of 0.485 (such as might be observed with 57% events on the placebo arm and 40% events on the TXA arm). Using the R package RCTdesign (or equivalently, S-Plus S+SeqTrial), a stopping boundary having three analyses at 50%, 75% and 100% of the planned sample size and with a level 0.2 O'Brien-Fleming boundary for harm would be computed using the RCTdesign code:

```
seqDesign(prob.model = "prop", null.hypothesis = 0.57,
    alt.hypothesis = 0.4, nbr.analyses = 3,
    sample.size = c(165, 248, 330), test.type =
"two.sided",
```

```
power = "calculate", alpha = c(0.025, 0.2),
P = c(Inf, 1))
```

For this example stopping boundary, the following table presents the critical values for harm and the corresponding adjusted statistical inference at each of the formal interim analyses. Critical values are expressed in terms of the crude estimate of treatment effect (TXA bleeding rate minus control bleeding rate), a Z statistic, and a one-sided fixed sample P value testing for harm. The adjusted statistical inference is a point estimate based on the bias adjusted mean and 95% confidence intervals and one-sided P values testing for harm using the likelihood ratio ordering of the outcome space.

Analysis		Stopping Boundaries			Adjusted Inference		
	N	Crude Estimate	Z	Fixed P val	Estimate	95% CI	One-sided P value
50%	165	0.111	1.440	.075	0.095	(-.004, .181)	0.123
75%	248	0.074	1.175	.120	0.059	(-.018, .132)	0.178
100%	330	0.055	1.019	.154	0.043	(-.022, .115)	0.200

Rationale: Owing to the need for adequate safety data, even if TXA is associated with markedly less bleeding than placebo, there is an imperative to gather safety data on the full sample size. Because TXA is currently being used off-label in the thrombocytopenic setting, it is important to document any harm due to TXA treatment, rather than just documenting the absence of a markedly beneficial effect.

3. Sample Size

Calculation of sample size and statistical power were made using S+SeqTrial based on the chi-squared test of association, which is equivalent to the score test from simple logistic regression. Based on 1:1 randomization, a one-sided level of significance 0.025, a design alternative hypothesis of 30% relative reduction in bleeding rates (57% on the placebo arm and 40% on TXA), a sample size of 330 subjects (165 TXA, 165 placebo) will provide 88% statistical power to declare statistical significance on the primary endpoint. This sample size will provide 74.8% statistical power to detect a 25% relative reduction in bleeding rates (57% vs 42.75%).

4. Precision of Inference for Efficacy

With the planned sample size and a placebo bleeding rate of 57%, an observed absolute decrease in WHO grade 2 or above bleeding of 10.6% (so 57% on placebo, 46.4% on TXA) would be judged statistically significant. Such an absolute difference in rates corresponds to a NNT of 9.4. If the baseline placebo bleeding rate were instead 40%, the threshold for statistical significance would be an absolute reduction of 10.14% (NNT = 9.9), and a baseline placebo rate of 70% would have a threshold of 10.35% (NNT = 9.7). These results are judged to be of the magnitude to possibly affect clinical practice, and allow for some added loss of precision with multiply imputed missing data.

5. Precision of Inference for Safety

This study is not powered to establish the definitive safety of the treatment with respect to the frequency of VTE. However, an observed difference in frequency of VTE of 3.7% on the placebo arm and 5.5% or less on the TXA therapy arm would result in a 95% confidence interval that excluded a relative risk of 3.0.

```
library(survival)
library(sandwich)

impSeed <- 20170615

# helper functions for below

na.locf <- function(vec){ # perform LOCF
  for (i in 2:length(vec)){
    vec[i] <- ifelse(!is.na(vec[i]),vec[i],vec[i-1])
  }
  return(vec)
}

makeLong <- function(dat){ # a function that shifts from
 wide (column for each event day) to long
  longdat <-
 data.frame(caseid=NULL,trt=NULL,site=NULL,group=NULL,start
 =NULL,end=NULL,bleed=NULL,count=NULL)
  nm <-
 c("caseid","trt","site","group","tstart","tend","bleed","c
 ount")
  for (i in 1:length(dat$caseid)){
    for (j in
 1:min(dat$daysonstudy[i],dat$eventday[i],na.rm=TRUE)){
      longdat <-
 rbind(longdat,c(dat$caseid[i],dat$trt[i],dat$site[i],dat$g
 roup[i],j-1,j,

 ifelse(is.na(dat$bleed)[i],0,

 ifelse(!is.na(dat$eventday[i]) & j==dat$eventday[i] &
 dat$bleed[i]==1,1,0)),
                                    dat[i,7+j])) # This may
 need to be changed to point to the right column in true
 data
    }
  }
  colnames(longdat) <- nm
  longdat$trt <- as.logical(longdat$trt)
  longdat$tstart <- as.numeric(longdat$tstart)
  longdat$tend <- as.numeric(longdat$tend)
  longdat$bleed <- as.numeric(longdat$bleed)
  longdat$count <- as.logical(longdat$count)
  return(longdat)
```

```
}

logisticScore <- function(datIn,est,nullHyp){ # compute
 the score and expected information for testing if trt==0
  x <- model.matrix(bleed ~ site + group + trt,data=datIn)
  y <- datIn[,"bleed"]
  nullest <- c(est,nullHyp) # we maximize under
 trt=nullHyp for est, then include trt=nullHyp for score
  nullp <- exp(x %*% nullest)/(1+exp(x %*% nullest)) # MLE
 under null hypothesis
  scorevec <- t(x) %*% (y-nullp)
  scorevar <- t(x) %*%
 diag(as.numeric(nullp)*as.numeric(1-
 nullp),nrow=length(nullp),ncol=length(nullp)) %*% x
  return(list(score=scorevec,info=scorevar))
}

treatScoreTest <- function(datImp,nullHyp){
  lmFit <- glm(bleed ~ site + group+offset(nullHyp*trt), #
 maximizes under some null hypothesis (default to coef=0)
      family="binomial",data=datImp)
  lmCoef <- coef(lmFit)
  scoreOut <- logisticScore(datImp,lmCoef,nullHyp)

  trtScore <- scoreOut[[1]][6] # treatment element of
 score
  trtVar <- scoreOut[[2]][6,6]-scoreOut[[2]][6,-6] %*%
 solve(scoreOut[[2]][-6,-6]) %*% scoreOut[[2]][-6,6] #
 variance of treatment element of score after estimating
 nuisance parameters
  return(c(trtScore,trtVar))
}


impute <- function(x,x.imp) ifelse(is.na(x),x.imp,x) #
 replace missing values of x with x.imp

# ------------------------------ Primary Efficacy
 Analysis ----------------------

### Expect a data frame structure with the following
 variables:
# caseid, treatment arm, casesite, therapeutic group, days
 on study (1-30),
```

```
# bleed (0=no bleed during 30 days, 1=bleed, NA=withdrew
 before 30 days), day of bleed
# 30 variables for each day of ANC count (high/low)

### Then we want to build "imps" number of data sets for
 each row with bleed = NA

### Note the long format described below has one row for
 each study day for each caseid, and variables:
# caseid, treatment arm, casesite, therapeutic group,
 tstart (0 to (daysonstudy-1)), tend(1 to daysonstudy),
# bleed (0, 1, NA), ANC count

### Imputation Scheme as Follows:
# 0. Reduce the data set to only complete case data and
 change to long format
# 1. Fit coxph model for each trt*stratum (estimate
 baseline hazard and coefficients) on the complete data
# 2. Repeat the following for each row with bleed = NA
 (consider row "i")
# 2a. Re-build the data set by removing all ANC columns
 after ith days on study
# 2b. Copy the final column to fill back to 30 columns of
 ANC
# 2c. Now use the data to estimate the hazard function for
 that subject by plugging into the appropriate trt*stratum
 coefficients
# 3. Use the resultant hazard functions to sample possible
 event times and repeat "imps" number of times
# 4. Combine the estimates from K = 20 imputation models
 with Wald test
# 5. Adjust the above fixed-sample test for sequential
 monitoring with RCTdesign



# Build the analysis dataset: step 0
actPt <- !is.na(studyJDates$activate) # Does not use the
 activation based on platelet
actBase <-
 data.frame(caseid=baseline$caseid[actPt],site=baseline$cas
 esite[actPt],group=baseline$stratum[actPt],

 activ=studyJDates$activate[actPt],daysonstudy=studyJDates$
```

```
effobstime[actPt],totalstudy=studyJDates$endstudy[actPt]-
studyJDates$activate[actPt],

stopReason=studyDates$studystopreason[actPt],deathCause=st
udyDates$textcausedeath[actPt])

actBase$trt <-
trtDF$trt[match(actBase$caseid,trtDF$caseid)] # use the
correct randomization

actBleeds <- bleeds[names(bleeds) %in% actBase$caseid] #
get the bleed object for activated cases
bleedCols <- data.frame(bleed=NULL,eventday=NULL) # set up
an object that will record first bleed 2+ and their dates

naInt <- vector(mode="list",length(actBleeds)) # will
store the intervals with missingness

for (i in 1:length(actBleeds)){
  bld <- actBleeds[[i]]
  bld2 <- (as.numeric(bld$maxBldgrade) >= 2) & (bld$stage
== "efficacy")

  availDts <- floor(bld$JDates)-actBase$activ[i]+1
  naDts <- (1:30)[!1:30 %in% availDts] # which days are
not available

  if (sum(bld2) > 0){
    bldInfo <- c(1,bld$JDates[min(which(bld2))]-
actBase$activ[i]+1) # there is a 2+ bleed
    naInt[[i]] <- NA # do not worry about missingness
  } else if (actBase$daysonstudy[i] == 30 &
length(naDts)==0) { # CHANGE: checks to make sure that
there are 30 bleed entries
    bldInfo <- c(0,NA) # 30 days of bleed entries and no
bleed
    naInt[[i]] <- NA # no missingness at all
  } else {
    bldInfo <- c(NA,min(naDts)) # someone to impute, give
the first day of missingness as "eventday"
    naInt[[i]] <- naDts
  }
  bleedCols <- rbind(bleedCols,bldInfo)
}
colnames(bleedCols) <- c("bleed","eventday")
```

```
names(naInt) <- actBase$caseid

actLabs <- labs[names(labs) %in% actBase$caseid] # get the
 labs for activated subjects
labCols <- NULL # object that will hold 30 days of ANC
for (i in 1:length(actLabs)){
  lb <- actLabs[[i]]$ANC
  dts <- floor(lb$JDates) - actBase$activ[i]+1 # ensures
 that ANC is counted on the day it happened, day 1 is all
 day after activation
  dtIdx <- !duplicated(dts) & (dts >= 1) & (dts <= 30) #
 takes the first  ANC measurement on each day in efficacy
 window
  cnt <- rep(NA,30) # initialize a vector of 30 days of
 ANC
  cnt[dts[dtIdx]] <- lb$value[dtIdx] <= 0.5 # low ANC <=
 500
  cnt <- na.locf(cnt) # LOCF
  labCols <- rbind(labCols,cnt)
}

# set up an analysis dataset dat
dat <-
 as.data.frame(cbind(actBase$caseid,actBase$trt,actBase[,c(
 2,3,5)],bleedCols,labCols)) # only keep relevant columns
colnames(dat) <-
 c("caseid","trt",colnames(actBase[,c(2,3,5)]),colnames(ble
 edCols),paste("day",1:30,sep=""))
longdatComp <- makeLong(dat[!is.na(dat$bleed),]) # makes
 ANC into a time-varying covariate for coxph

# Begin the imputation procedure: step 1
bHaz <- function(trt,longdat){
  dat <- longdat[longdat$trt==trt,]
  lm <-
 with(dat,coxph(Surv(time=tstart,time2=tend,event=bleed)~si
 te+count+cluster(caseid)+strata(group))) # cox model with
 time-varying ANC and robust SE
  return(lm)
}

lmPlc <- bHaz(0,longdatComp) # get the baseline hazard in
 placebo and control groups
lmTrt <- bHaz(1,longdatComp)
bList <- list(lmPlc,lmTrt)
```

```
oneimp <- function(bList,datInc){ # the function that
creates one imputed dataset and outputs the score test on
treatment parameter
  naCases <- is.na(datInc$bleed) # get the cases with
missing bleed
  for (i in 1:sum(naCases)){
    naDay <- datInc$eventday[naCases][i]-1 # the last day
with recorded data
    naTrt <- datInc$trt[naCases][i]
    naCase <- datInc$caseid[naCases][i] # grab the caseid

    # step 2
    truncData <- datInc[,1:(7+naDay)] # take the columns
up until the last day of data
    frzCol <- datInc[,(naDay+7)]
    frzData <- cbind(truncData,matrix(rep(frzCol,30-
naDay),ncol=30-
naDay,nrow=dim(datInc)[1]))[which(naCases)[i],] # choose
the subject of interest
    impData <- makeLong(frzData)

    lmNum <- ifelse(naTrt==0,1,2)
    lm <- bList[[lmNum]] # returns appropriate baseline lm

    # step 3
    predSurvival <- survfit(lm,impData) # predict survival
based on imputed data

    empCDF <- 1-as.matrix(predSurvival$surv)[,1] #
survival estimates for 30 days
    invCDF <- function(prob) { # takes in a probability
and returns the smallest t (# of days) to achieve that
probability
      min(which(empCDF >= prob)) # specific eCDF
    }
    u <- runif(1)
    t <- invCDF(u)

    missItvl <- naInt[[match(naCase,names(naInt))]]
    iBleed <- ifelse(t %in% missItvl,1,0) # if bleed would
have occurred in 30 days (after time we actually observed)
then fill in
    datInc$bleed[naCases][i] <- iBleed
  }
```

```
  return(datInc)  # return the filled in data set
}

# step 4
set.seed(impSeed)
reps <- 20

impDatasets <- vector("list",reps)
bleedLogistImps <- bleedLogistImpsMLE <-
 matrix(nrow=reps,ncol=2)
for (k in 1:reps){
  impOut <- oneimp(bList=bList,dat=dat)
  impDatasets[[k]] <- impOut # save the imputed data sets
 for use in interval estimation
  impFit <- glm(bleed ~ site + group+trt,
                family="binomial",data=impOut) # the full
 model fit on the imputed data
  bleedLogistImps[k,] <- treatScoreTest(impOut,0)
  bleedLogistImpsMLE[k,] <-
 c(coef(impFit)[6],sandwich(impFit)[6,6])
}
bleedLogistFixed <- mean(bleedLogistImps[,1]) /
 sqrt(mean(bleedLogistImps[,2]) + var(bleedLogistImps[,1]))
 # wald-imputed test statistic
bleedLogistFixed.p <-
 pnorm(bleedLogistFixed,lower.tail=FALSE) # one-sided,
 upper wald p-value from imputation

bleedLogistFixedOR <- exp(mean(bleedLogistImpsMLE[,1]))

# step 5
source("finalRCTDesign.R") # this should output a final
 seqDesign object 'atreatFinal'

# bleedLogistSeq <-
 seqInference(atreatFinal,observed=bleedLogistFixed, #
 convert the fixed-sample wald statistic to sequential
 monitor
                                  #
 analysis.index=3,ordering="l",inScale="Z") # use the LR
 ordering

bleedLogistSeq <-
 seqInference(atreatFinal,observed=bleedLogistFixed.p, #
 convert the fixed-sample p-value to sequential monitor
```

```
analysis.index=3,ordering="l",inScale="P") # use the LR
ordering

bleedLogistSeq.p <-
2*min(bleedLogistSeq$PvalueL.LROrder,bleedLogistSeq$Pvalue
U.LROrder) # two-sided final p-value

digOR <- 2
digP <- 3

print(paste0("OR=",round(bleedLogistFixedOR,digOR),
             " P=",round(bleedLogistSeq.p,digP))
      )



############################################################
############################
# Above code computed the multiple imputation- and
sequential monitoring-adjusted p-value and multiple
imputation-adjusted OR
# Code below computes the multiple imputation- and
sequential monitoring-adjusted confidence interval

# 1. Function to compute the multiple imputation- and
sequential monitoring-adjusted p-value for different
"null" hypotheses
# 2. Use the multiple imputation-adjusted Wald CI to guide
a grid search across different "null" hypotheses
#     want to collect the points at which the lower
endpoint switches from <0.025 to >0.025 and upper switches
from <0.975 to >0.975


alphaLevel <- 0.05
lowerCrit <- alphaLevel/2
upperCrit <- 1-lowerCrit


### Step 1: Repeat much of the same code as above to
compute the multiple imputation- and sequential
monitoring-adjusted p-value
```

```
oneP <- function(nullP,impDataList=impDatasets){ # needs a
"null" to test at and the list contaning imputed datasets
  ## Changes to above code: do not need to return MLE,
returns 2-sided p-value rather than 1-sided

  bleedLogistImps <-
matrix(nrow=length(impDataList),ncol=2)
  for (k in 1:length(impDataList)){
    bleedLogistImps[k,] <-
treatScoreTest(impDataList[[k]],nullP)
  }
  bleedLogistFixed <- mean(bleedLogistImps[,1]) /
sqrt(mean(bleedLogistImps[,2]) + var(bleedLogistImps[,1]))
# wald-imputed test statistic
  bleedLogistFixed.p <-
pnorm(bleedLogistFixed,lower.tail=FALSE) # one-sided,
upper wald p-value from imputation

  # step 5
  bleedLogistSeq <-
seqInference(atreatFinal,observed=bleedLogistFixed.p, #
convert the fixed-sample p-value to sequential monitoring

analysis.index=3,ordering="l",inScale="P") # use the LR
ordering

  bleedLogistSeq.p <-
2*min(bleedLogistSeq$PvalueL.LROrder,bleedLogistSeq$Pvalue
U.LROrder) # two-sided final p-value

  return(bleedLogistSeq.p)
}



### Step 2a: compute the fixed sample CI as a coarse guide
 for the final CI grid search

bleedLogistFixedLogOR <- mean(bleedLogistImpsMLE[,1])
bleedLogistFixedLogORSE <-
 sqrt(var(bleedLogistImpsMLE[,1])+mean(bleedLogistImpsMLE[,
 2]))

coarseCI <-
 bleedLogistFixedLogOR+qnorm(c(lowerCrit,upperCrit),0,1)*bl
```

```
eedLogistFixedLogORSE # coarse CI (NOTE: this is on the
parameter or log(OR) scale!)

### Step 2b: function to perform grid search given initial
values

gridSearch <- function(inits,tol,lower=TRUE){ # requires
initial guess and tolerance, lower versus upper changes
the inequalities due to ordering of parameter values; uses
global variable of significance levels
  if (lower){ # moves from rejecting "null" to accepting
    seqRange <- seq(inits[1],inits[2],length.out=1/tol)
    seqP <- rep(NA,length(seqRange))
    for (i in 1:length(seqRange)){
      seqP[i] <- oneP(seqRange[i]) # compute the p-value
with the given "null" hypothesis
      if (i==1 & seqP[i] >= alphaLevel) {
        stop("Need smaller lower guess") # no switch
possible in the given range
      } else if (i==length(seqRange) & seqP[i] <
alphaLevel){
        stop("Need larger upper guess") # no switch
happened in the given range
      } else {
        if (i > 1){
          if (seqP[i] >=alphaLevel & seqP[i-1] <
alphaLevel){
            switchVals <- c(seqRange[i-1],seqRange[i]) #
stores the points at which it switches significance
            break # found the switching point!
          }
        }
      }
      # if (i %% 5==0) print(i)
    }
  } else { # moves from accepting null to rejecting
    seqRange <- seq(inits[1],inits[2],length.out=1/tol)
    seqP <- rep(NA,length(seqRange))
    for (i in 1:length(seqRange)){
      seqP[i] <- oneP(seqRange[i]) # compute the p-value
with the given "null" hypothesis
      if (i==1 & seqP[i] < alphaLevel) {
        stop("Need smaller lower guess") # no switch
possible in the given range
```

```
      } else if (i==length(seqRange) & seqP[i] >=
alphaLevel){
        stop("Need larger upper guess") # no switch
happened in the given range
      } else {
        if (i > 1) {
          if(seqP[i] < alphaLevel & seqP[i-1] >=
alphaLevel){
            switchVals <- c(seqRange[i-1],seqRange[i]) #
stores the points at which it switches significance
            break # found the switching point!
          }
        }
      }
      # if (i %% 5==0) print(i)
    }
  }
  return(switchVals)
}

computeCI <-
function(inits=c(coarseCI[1],coarseCI[2]),tolFine=1e-
3,tolCoarse=1e-1,coarseFactor=1){ # requires initial CI
guess, and tol is how expensive of a search; tol must be
<1e-2

  ### compute lower bound: when does it cross from <0.025
to >= 0.025
  coarseLowerInits <- inits[1]+c(-
1,1)*diff(inits)*coarseFactor
  coarseLowerSwitch <-
gridSearch(coarseLowerInits,tolCoarse,lower=TRUE)
  fineLowerSwitch <-
gridSearch(coarseLowerSwitch,tolFine,lower=TRUE)


  ### compute upper bound: when does it cross from <=
0.975 to >0.975
  coarseUpperInits <- inits[2]+c(-
1,1)*diff(inits)*coarseFactor
  coarseUpperSwitch <-
gridSearch(coarseUpperInits,tolCoarse,lower=FALSE)
  fineUpperSwitch <-
gridSearch(coarseUpperSwitch,tolFine,lower=FALSE)
```

```
  # return(c(coarseLowerSwitch[1],coarseUpperSwitch[2])) #
 return the limits that are barely rejected by hypothesis
 tests
  return(c(fineLowerSwitch[1],fineUpperSwitch[2])) #
 return the limits that are barely rejected by hypothesis
 tests
}

coarseTol <- 1e-1
fineTol <- 1e-3
coarseWidthFactor <- 1

bleedLogistSeq.CI <-
 computeCI(tolCoarse=coarseTol,tolFine=fineTol,coarseFactor
 =coarseWidthFactor) # final confidence interval

print(paste0("95% CI:
 (",round(exp(bleedLogistSeq.CI[1]),3),",
 ",round(exp(bleedLogistSeq.CI[2]),3),")"))
```