Paper 154-29

Empirical Power Computation Using SAS® for Schuirmann's Two One-Sided Tests Procedure in Clinical Pharmacokinetic Drug-Drug Interaction Studies

Jihao Zhou, Bruce Coate and Yulei He*
Pfizer Global Research and Development
La Jolla Laboratories, San Diego, California
* The University of Michigan, Ann Arbor

ABSTRACT

Drug-drug interaction studies are becoming increasingly crucial in drug development. An FDA survey of recent new drug application submissions showed that 70% of drug-drug interaction studies were conducted using fixed-sequence designs. Schuirmann's Two One-sided Tests (TOST) procedure is the FDA preferred statistical method for evaluating drug-drug interaction. However, calculating the exact power of the TOST procedure generally requires sophisticated numerical integration. Monte Carlo simulation is a good alternative approach to the power calculation if a closed algerbraic formula is not available. This paper presents a SAS Macro to compute the empirical power of the TOST procedure under a fixed-sequence design. Empirical power curve plot and numerical table can be produced through this SAS macro.

Keywords: drug-drug interaction; empirical power; fixed-sequence design; Monte Carlo simulation; Two One-Sided Tests

INTRODUCTION

Drug-drug interaction is a key issue in clinical practice, especially for HIV/AIDS teatment, which entails the use of a combination of different classes of antiretroviral drugs (called cocktail therapy). More and more drug-drug interaction studies are needed in new drug applications (NDAs). In an US FDA survey of recent NDAs, 70% of clinical drug-drug interaction studies were conducted as a fixed sequence design [1]. Schuirmann's Two One-sided Tests (TOST) procedure [2] is the preferred statistical method for evaluating a drug-drug interaction effect based on drug systemic exposure as assessed by AUC and Cmax. However, calculating the exact power of the TOST procedure requires sophisticated numerical integration [2,6,7]. The Monte Carlo simulation is a good alternative approach to the power calculation. SAS has been used as a comuputing tool to perform such tasks [3, 4]. In this paper, a SAS macro is presented that can be used to compute the empirical power of the TOST procedure in a fixed-sequence design through Monte Carlo simulations.

BACKGROUND

PHASE 1 DRUG-DRUG INTERACTION TRIALS IN A FIXED-SEQUENCE DESIGN

A Phase 1 drug-drug interaction study is generally considered for drugs that are likely to be administered concomitantly [5]. A drug involved in drug-drug interactions is classified as either a precipitant drug (a drug that causes a change in pharmacokinetics of another drug) or an object drug (a drug that is affected by the precipitant drug). The pharmacokinetic parameters of the object drug, such as AUC and Cmax, will be compared in the presence and absence of the precipitant drug. In a fixed-sequence design, treatments are administered sequentially over two periods. The object drug is administered alone during the first period, followed by a washout period, and then the object and precipitant drugs are administered concomitantly in the last period.

SCHUIRMANN'S TWO ONE-SIDED TESTS PROCEDURE

Let μ_d be the mean difference of the bioavailabilities, such as AUC and Cmax, of the object drug between the first and last periods depicted as above, θ_L denotes the lower no-effect boundary, and θ_U denotes the upper no-effect boundary, then the objective of the drug-drug interaction in a fixed-sequence design can usually be tested in the following two one-sided hypotheses:

Null hypothesis (H_0): $\mu_d \le \theta_L$ or $\mu_d \ge \theta_U$
Alternative hypothesis (H_a): $\theta_L < \mu_d < \theta_U$

Assuming logarithmically transformed data would be normally distributed, then H_0 is rejected at significance level α and no drug-drug interaction is concluded if 100 (1-2α) % CI $(\hat{\mu}_d - t_{1-\alpha,n-1}\hat{se},\ \hat{\mu}_d + t_{1-\alpha,n-1}\hat{se})$ of the mean μ_d is entirely within (θ_L, θ_U), otherwise H_0 fails to be rejected. The choices of no-effect boundaries depend on the specific drugs involved in the study. The commonly used θ_L and θ_U are (80%, 125%) and (77%, 130%). The type-I error α of the TOST procedure is often set as 5%.

There is no closed algebraic formula to calculate the statistical power for the TOST procedure. In cross-over designs, Muller [6], Phillips [7], and Schuirmann [2] used a numerical integration approach, which is complicated to implement in practice, while Chow and Liu [8] suggested using the Monte Carlo simulation to obtain the empirical power. We adopted the latter method to estimate the power for the TOST procedure in a fixed-sequence design.

METHODS

STATISTICAL SIMULATIONS

The Monte Carlo simulation involves random sampling techniques to generate a series of random samples from a distribution that represents the study population of interest (e.g., the population under H_a, or there exists no drug-drug interaction) [3]. For each generated random sample, the TOST procedure is applied and the conclusion of rejecting or accepting H_0 is made. The empirical power of the TOST procedure is then calculated as the proportion of the replications in which H_0 is rejected. The computation process is outlined as follows:

Step 1. Generate a random sample of size n according to the following distribution with pre-specified μ_d and σ^2,

$$X \sim N(\mu_d, \sigma^2)$$

Step 2. Calculate the 100 (1-2α) % CI for μ_d,

Step 3. Repeat steps 1 and 2 N times, e.g., N=1000,

Step 4. The empirical power is calculated as the proportion of N random samples in which the confidence interval (CI) falls entirely within pre-specified (θ_L, θ_U).

SAS PROGRAM DEVELOPMENT

THE SAS MACRO CORE_POWER (ATTACHED)

First, the macro deletes data set ci_total if any exists from previous macro calls. The SAS RANNOR() function generates random values from a normal distribution with mean 0 and standard deviation (SD) 1. In a fixed-sequence design, the SD of the log difference approximately equals the coefficient of variation (CV) times the square root of 2. Multiplying the resulting value from RANNOR() by the SD and adding the mean difference generates random numbers for the distribution under H_a. For each simulation, PROC MEANS is used to calculate 100 (1-2α) % CI for the mean difference and save it into a data set samp_tost. PROC APPEND is then used to concatenate all the resulting data sets. Another SAS data step is used to produce a binary (0/1) variable to indicate whether the CI from each simulation falls within the criteria (θ_L, θ_U). If it does, it means the H_0 is rejected. SAS PROC MEANS is then used to calculate the average of the binary indicator variables to obtain the proportion of rejecting H_0 for all simulations (i.e., the empirical power of the TOST procedure).

THE SAS MACRO POWER (ATTACHED)

To obtain the distribution plot of the empirical power spanning the parameter space (θ_L, θ_U), setting the macro parameter PLOT=yes calls the CORE _POWER to plot the power curve. By default, 15 equally spaced points are selected from (θ_L, θ_U) including zero and two boundary points. PROC GPLOT is used to create the power curve. In addition, setting the parameter TABLE=yes produces a numerical power table for a certain range of commonly used sample sizes, intra-subject CVs, and no-effect boundaries. PROC REPORT and ODS are used to create the table.

RESULTS

In designing clinical trials, usually at least 80% power is required. Hence a reference line indicating 80% power is drawn in the power plot. Figure 1 shows that the power curve is symmetric and it achieves the maximum empirical power when the mean difference is zero. In practice, we may also need to estimate the power at a small non-zero mean difference, such as 0.025, to get a relatively conservative power estimate. Table 1 presents the empirical power estimates for commonly used sample sizes, CVs and no-effect boundaries. The Monte Carlo simulation involves random sampling techniques to generate a series of random samples from a distribution that represents the study population.

Figure 1. Empirical Power Curve Based on 1000 Simulations

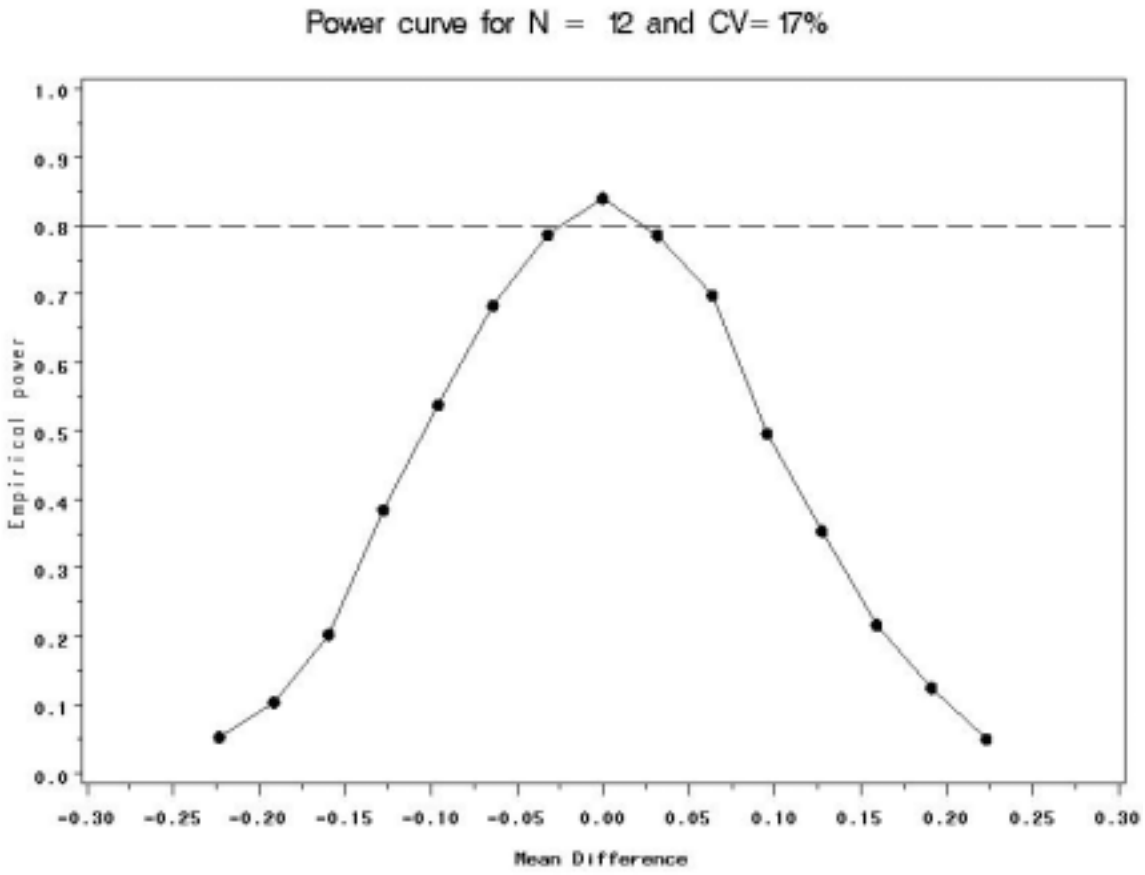

Table 1. Empirical Power (%) for Common Clinical Pharmacokinetic Drug-Drug Interaction Studies With $\mu_d = 0$ Based on 1000 Simulations

		CV (%)						
Criteria	**N**	**10**	**15**	**20**	**25**	**30**	**35**	**40**
(0.77, 1.30)	8	0.995	0.863	0.521	0.285	0.131	0.047	0.026
	10	1.000	0.946	0.737	0.410	0.216	0.082	0.040
	12	1.000	0.985	0.798	0.553	0.321	0.136	0.060
	14	1.000	0.991	0.905	0.677	0.449	0.212	0.114
(0.80, 1.25)	8	0.981	0.700	0.343	0.138	0.057	0.019	0.006
	10	0.998	0.871	0.496	0.246	0.093	0.041	0.009
	12	1.000	0.934	0.651	0.344	0.142	0.063	0.023
	14	1.000	0.958	0.737	0.479	0.231	0.084	0.038

CONCLUDING REMARKS

The Monte Carlo simulation is a useful approach for estimating the power of a variety of statistical tests [4], especially when there are no closed formulae for power calculation. SAS is a practical tool for carrying out simulation [3,4]. Here we present a SAS macro to estimate the power of the Schuirmann's TOST procedure in a fixed-sequence design for clinical drug-drug interaction trials. In future studies, we will implement similar simulation procedures to estimate the power for higher-order crossover designs.

REFERENCES

1. Huang S-M, Lesko LJ, and Williams RL. Assessment of the quality and quantity of dug-drug interaction studies in recent NDA submissions: Study design and data analysis issues. *J of Clin Pharmacology* 1999; 39: 1006-1014.

2. Schuirmann DJ. A Comparison of the Two One-Sided Tests Procedure and the Power Approach for Assessing the Equivalence of Average Bioavailability. *J of Pharamcokinetics and Biopharmaceutics* 1987; 15(6): 657-680.

3. Fan X, Felsovalyi A., Sivo SA, and Keenan SC. 2002. *SAS® for Monte Carlo Studies: A guide for Quantitative Researchers*. Cary, NC: SAS Institute Inc. pp 66-71.

4. Graebner RW. Study Design with SAS: Estimating Power with Monte Carlo Methods.*2002 SUGI 27 Proceedings*, Paper 265-25.

5. FDA Guidance for Industry. *In Vivo Drug Metabolism/Drug Interaction Studies – Study Design, Data Analysis, and Recommendations for Dosing and and labeling..* FDA, Center for Drug Evaluation and Resaerch (CDER) and Center for Biological Evaluation and Resaerch (CBER), November 1999.

6. Muller-Cohrs J. The Power of the Anderson-Hauck Test and the Double T-Test. *Biom. J* 1990; 32(3): 259-266.

7. Phillips K. Power of the Two One-Sided Tests Procedures in Bioequivalence. *J of Pharamcokinetics and Biopharmaceutics* 1990; 18(2): 137-144.

8. Chow S-C and Liu J-P. *Design and Analysis of Bioavailability and Bioequivalence Studies*. Markel Dekker, Inc., New York, NY. 2000.References go at the end of your paper.

ACKNOWLEDGMENTS

We are grateful to Glen Andrews and Alex Kuperman at Pfizer Inc and Melissa Chen at Prosanos Corporation for valuable suggestion in preparation of this paper.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

J. Zhou and B. Coate can be reached at Pfizer Inc, 11085 Torreyana Road, San Diego, CA 92121. Y. He can be reached at the University of Michigan, 1586 Murfin Avenue #37, Ann Arbor, MI 48105.

ATTACHMENTS

THE MACRO POWER:

```
/***------------------------------------------------------------------------***
SAS Macro: POWER.sas;
Function:  Compute empirical power of Schuirmann's TOST for crossover like design
Remarks:   SAS 8.2 for Windows
***------------------------------------------------------------------------***/
options sasautos=('C:\temp\');
options nosymbolgen nomlogic nomprint nodate nonumber;

/***------------------------------------------------------------------------***
 Definition of the parameters in Macro POWER:
    SampeN= sample size
    CV= coefficient of variation in percentage such as 15
    Alpha= type-I error, by default, it uses 0.05
    Runs=  number of the replication for simulation, suggest minimum = 1000
```

```
    Lower= lower criteria boundary of population mean difference= 0.80
    Upper= upper criteria of population mean difference, default= 1.25
    meand= the mean difference at which power is calculated, default is zero
    plot=  yes/no to make or not to draw empirical power curve
    table= yes/no to present or not present the numeric power in table
***-------------------------------------------------------------------------***/

%Macro POWER(plot=no, table=yes, sampleN=12, CV=15, alpha=0.05, runs=1000,
             lower=0.80, upper=1.25, meand=0, npoints=15);

* Delete dataset empower if any ***;
%if %sysfunc(exist(empower)) %then
    %do;
    Proc datasets library=work nolist;
    delete empower;
    run; quit;
    %end;

%if %upcase(&plot)=YES %then
%do;
    * The Macro Call to Make Power Curve ***;
    %do number=1 %to &npoints;

    * create meand for making plots;
    %let plot_mean=log(&lower) + (&number - 1) * (log(&upper) - log(&lower)) /
(&npoints - 1);
    data _null_;
    y=&plot_mean;
    call symput('meand',y);
    run;

    %CORE_POWER(meand=&meand, runs=&runs, sampleN=&sampleN, CV=&CV, lower=&lower,
                upper=&upper, alpha=&alpha)
    %end;

    goptions reset=all;
    symbol1 interpol=line value=dot height=1 c=black;
    legend across=1
           cborder=blue
              position=(top inside right)
              mode=share
              ;
    legend2;
    axis1 label=('Mean Difference' height=2.5 ) order=(-0.30 to 0.30 by 0.05)
              minor=(number=1);
    axis2 label=(angle=90 'Empirical power' height=2.5) order=(0.0 to 1.0 by 0.1)
              minor=(number=1);

    proc gplot data=empower;
           plot power_x * meand/ haxis=axis1 vaxis=axis2 vref=0.80 lvref=2 ;
    title height=1.5 "Power curve for N = &sampleN and CV=&CV.%";
    run; title; quit;
%end;

%if %upcase(&table)=YES %then
    %do;
    %if %sysfunc(exist(empower)) %then
                  %do;
                  Proc datasets library=work nolist;
                         delete empower;
                  run; quit;
                  %end;

* execute from n=12 to 20 by 2;
```

```
    %do n=12 %to 20 %by 2;
           %do CV=10 %to 40 %by 5;
        %CORE_POWER(meand=&meand, runs=&runs, sampleN=&n, CV=&CV, lower=&lower,
                    upper=&upper, alpha=&alpha)
           %end;
    %end;

* make the report;
/**data power_all **/
data power_all;
  set empower;
  lcrit=round(lower,0.01);
  ucrit=round(upper,0.01);
  crit="("||trim(left(put(lcrit,4.2)))||", "||trim(left(put(ucrit,4.2)))||")";
  _n=input(n,best.);
run;

proc sort data=power_all out=power_all;
  by lower _n cv;
run;

proc transpose data=power_all out=xpower(where=(_name_="power_x"));
  by lower _n ;
  var power_x;
  id cv;
  copy crit;
run;

ods listing close;

ods rtf file='C:\temp\table.rtf';

options nodate nonumber orientation=portrait;

proc report data=xpower nowindows headline headskip spacing=2 split="@" ;
  columns crit _n ("CV (%)"  _10 _15 _20 _25 _30 _35 _40);
  define crit /"Criteria" width=16  order;
  define _n /"N" width=5 order=internal;
  define _10 /"10" width=5 f=5.3;
  define _15 /"15" width=5 f=5.3;
  define _20 /"20" width=5 f=5.3;
  define _25 /"25" width=5 f=5.3;
  define _30 /"30" width=5 f=5.3;
  define _35 /"35" width=5 f=5.3;
  define _40 /"40" width=5 f=5.3;
    break after crit/skip;
title; footnote;
run;
ods rtf close;
ods listing;

%end;

%Mend POWER;


*-------Example: Run the Macro to make numerical power table -------*;
%POWER (plot=no,
          table=yes,
          sampleN=12,
          CV=17,
          alpha=0.05,
          runs=5,
          lower=0.80,
```

```
            upper=1.25,
            meand=0,
            npoints=15)

* The End of This SAS Program;
```

THE MACRO CORE_POWER: (C:\TEMP)

```
/***-------------------------------------------------------------------------***
SAS Macro: CORE_POWER.sas
Function:  Compute the empirical power of Schuirmann's TOST under a cross -over like
           design. It is called in the macro POWER
***-------------------------------------------------------------------------***/
/***-------------------------------------------------------------------------***
 Definition of the paramers in macro CORE_POWER:
    SampeN= sample size
    CV     = coefficient of variation in percentage such as 15
    Alpha  = type-I error, by default, it uses 0.05
    Runs   = number of the replication for simulation, suggest minimum = 1000
    Lower  =l ower criteria boundary of population mean difference= 0.80
    Upper  = upper criteria of population mean difference, default= 1.25
    Meand  = the mean difference at which power is calculated, default is zero
***-------------------------------------------------------------------------***/
%Macro CORE_POWER(sampleN=&sampleN, CV=&CV, alpha=&alpha, runs=&runs, lower=&lower,
                  upper=&upper, meand=&meand);

* Delete dataset ci_total if any ***;
%if %sysfunc(exist(ci_total)) %then
    %do;
    Proc datasets library=work nolist;
           delete ci_total epower;
    run; quit;
    %end;

* Simulation starts ***;
* The SD of the log difference is the root 2 times CV;
%do rep=1 %to &runs;
    * Normal sample generation ***;
    data sample(keep=x);
           do i=1 to &sampleN;
          z=rannor(0);
          x=&meand + sqrt(2)*&CV/100*z;
           output;
           end;
    run;

    * Calculate the 90% CI for the mean and save into dataset samp_tost;
    proc means data=sample clm alpha=%sysevalf(2*&alpha) noprint;
           var x;
           output out=samp_tost lclm=t_low uclm=t_up;
           * Note: t_low/t_up denotes lower/upper CI boundary for TOST;
    run;

    * Concatenate the resulting CI datasets into CI_total;
    proc append base=ci_total data=samp_tost force;
    run;
%end;


* Create an indicator variable for empirical power ***;
data prop;
    set ci_total;
    * produce an indicator of empirical power;
```

```
    * if indi=1, reject the null hypothesis, otherwise not;
    indi=(t_low > log(&lower) and t_up < log(&upper));
run;

* Calculate empirical power ***;
proc means data=prop noprint;
    var indi;
    output out=propall(drop=_type_ _freq_) mean=power_x;
run;

* Incorporate passed parameters into the final dataset epower;
data epower;
    set propall;
    n=symget('sampleN');
    lower=symget('lower');
    upper=symget('upper');
    CV=symget('CV');
    runs=symget('runs');
    meand=1.0*symget('meand');
    alpha=symget('alpha');
run;

* Test the existence of the epower and do the appending;
%if %sysfunc(exist(empower)) %then
    %do;
    * append the data;
    proc append base=empower data=epower force;
    run;
    %end;
%else %do;
    proc datasets library=work nolist;
    change epower = empower;
    run;
    quit;
    %end;
%mend CORE_POWER;

* The end of this SAS Macro;
```