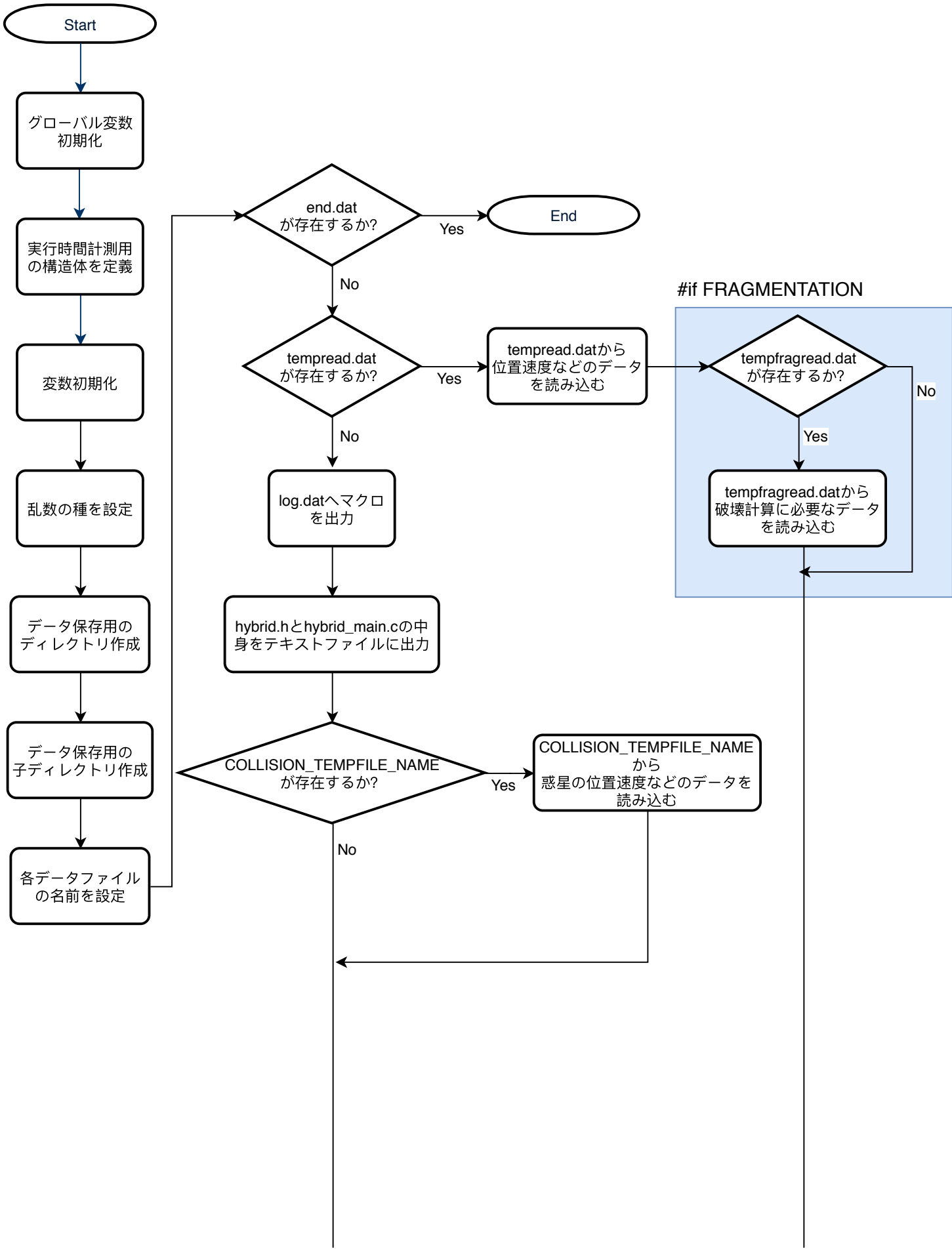
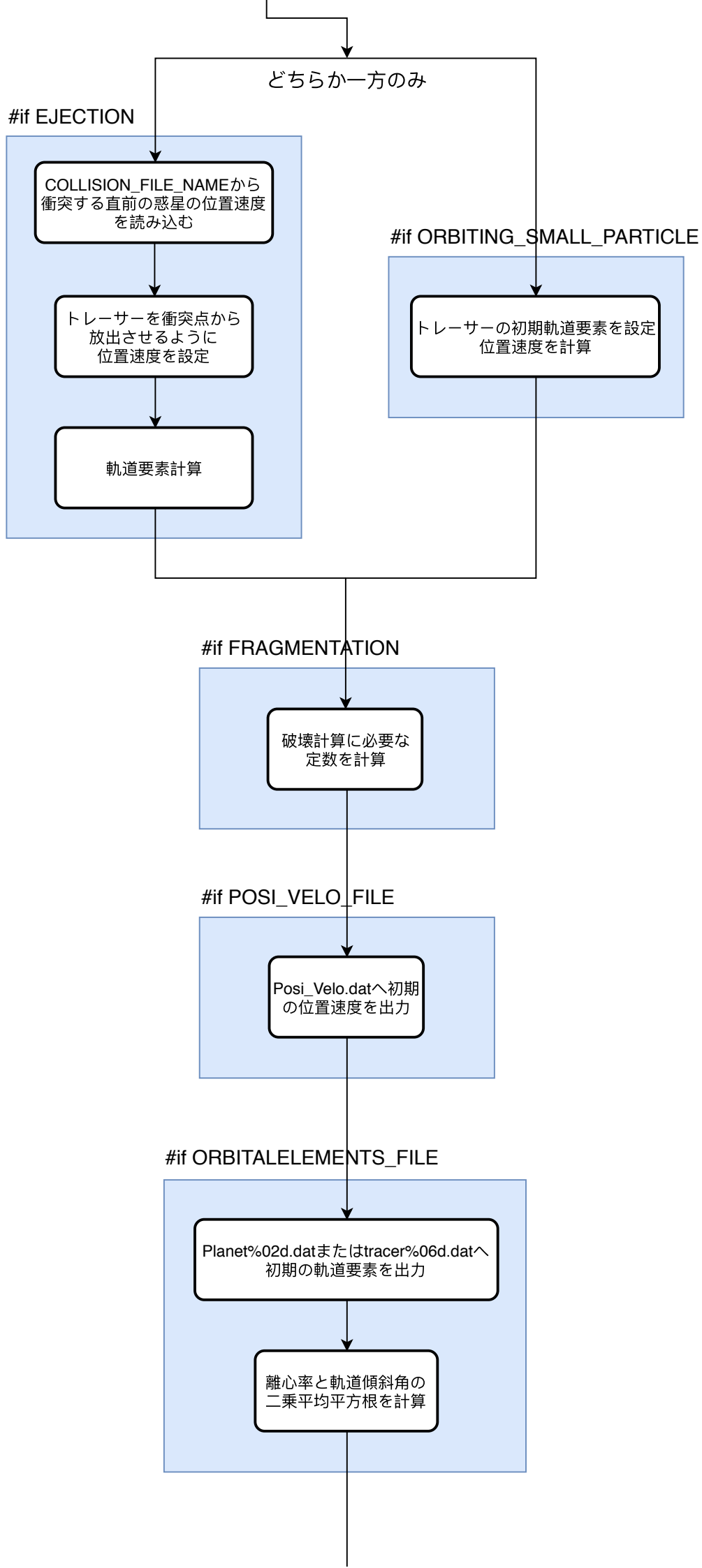


ISOYAforSaturn : 巨大衝突した天体から破片を放出させる計算

hybrid_main.c





#if ENERGY_FILE

中心星を含めた
重心計算

初期全エネルギー計算

初期全角運動量の
大きさ計算

Energy.datへエネルギー
と角運動量を出力

初期加速度と加加速度
を計算

初期タイムステップ計算
2進数に設定

ヒープソートを用いタイムステップが
小さな順に粒子を並び替え
index[]に並び替えた粒子番号を格納

タイムステップが一番
小さい粒子集団の総数
をn_i_sysとする

タイムステップが一番
小さい粒子集団のうち
一番若い粒子番号を
i_sysとする

基準とする惑星の軌道長半径から
 $\pm 0.5 \cdot \text{DELTA_HILL} \cdot$
(相互ヒル半径) の範囲にいる粒子の総
数をcenterとする
それより内側にいる粒子の総数をinner、
外側にいる粒子の総数をouterとする

#if TRACERLIST_FILE

tracerlist.datへ初期
に各粒子がどの範囲
にいるか出力

tracerlistnumber.dat
へinner, center,
outerの数を出力

#if FRAGMENTATION

近傍粒子探索 &
質量フラックス計算

inner, center, outerの
範囲の総質量と面密度
を計算

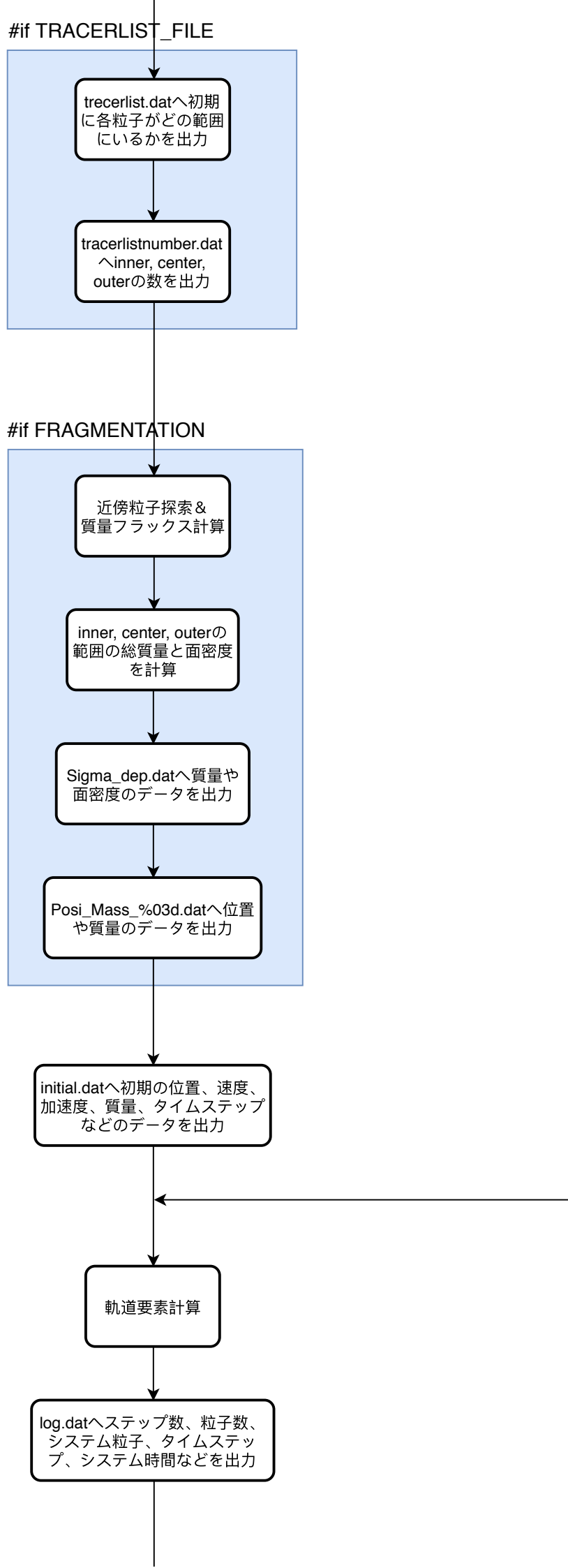
Sigma_dep.datへ質量や
面密度のデータを出力

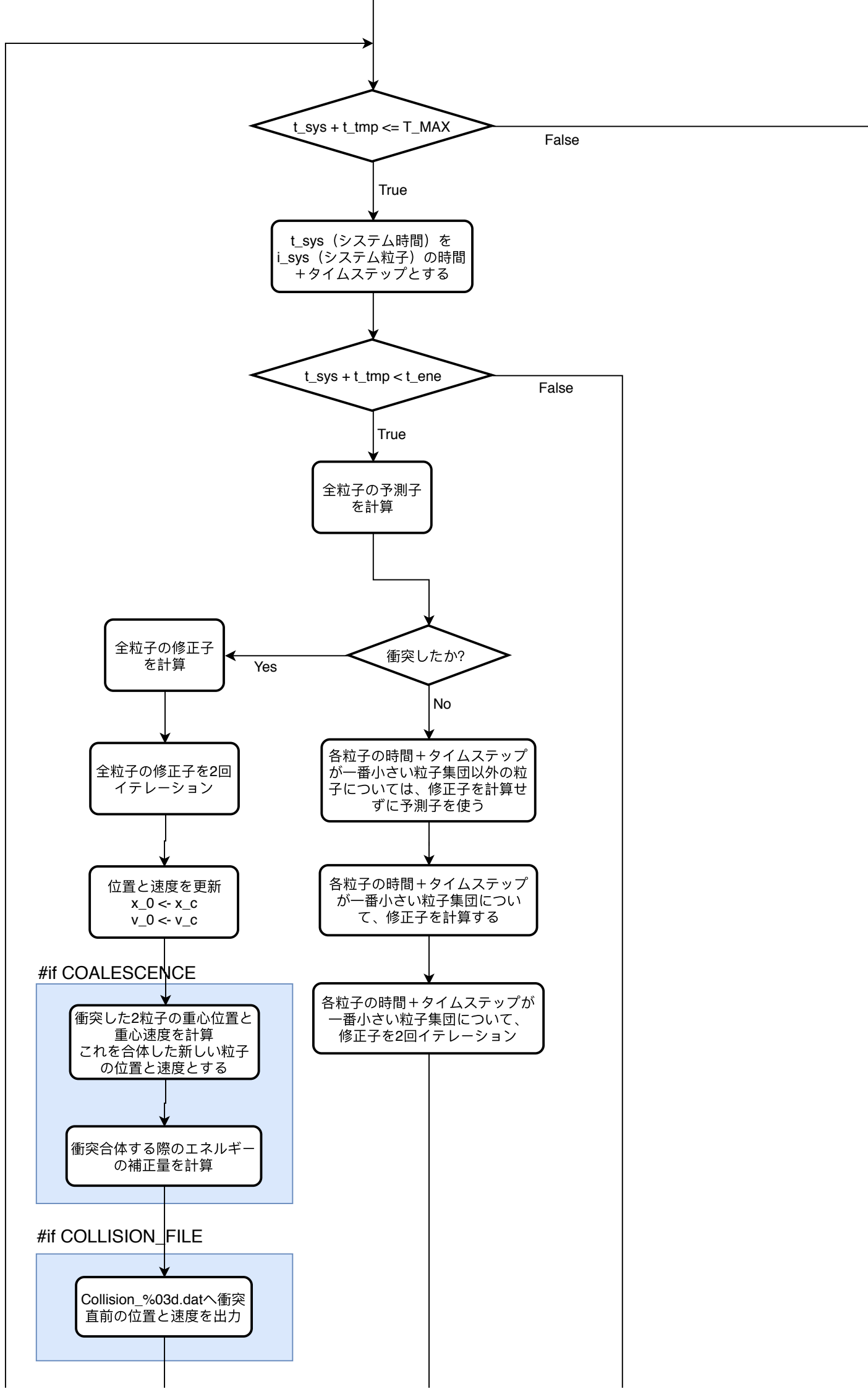
Posi_Mass_%03d.datへ位置
や質量のデータを出力

initial.datへ初期の位置、速度、
加速度、質量、タイムステップ
などのデータを出力

軌道要素計算

log.datへステップ数、粒子数、
システム粒子、タイムステッ
プ、システム時間などを出力





#if COALESCENCE

衝突した2粒子のうち、粒子番号が若い方を新しい粒子とし、もう一方を最後尾に持っていき以後ループから外す

全ての粒子の軌道要素計算

#if ENERGY_FILE

中心星を含めた
重心計算

全エネルギー計算

全角運動量の大きさ計算

Energy.datへエネルギー
と角運動量を出力

全粒子の加速度と
加加速度を計算

#if COLLISION_FILE

Collision_temp_%03d.datへ衝突
直後の位置と速度を出力

惑星同士の衝突か?

Yes

No

各粒子のタイムステップ
を、各粒子の時間から
t_eneまでの間隔とする

各粒子の時間を0に
リセットする

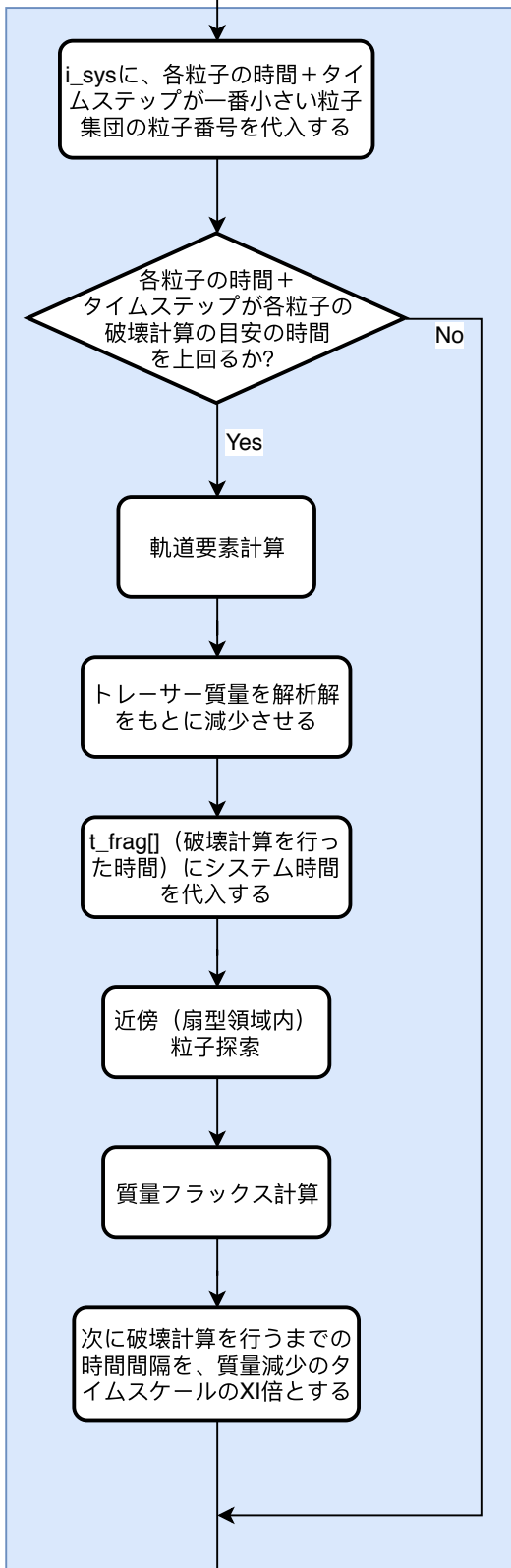
t_tmp <- t_tmp + t_sys
t_sys <- 0.0

全ての粒子の予測子
を計算する

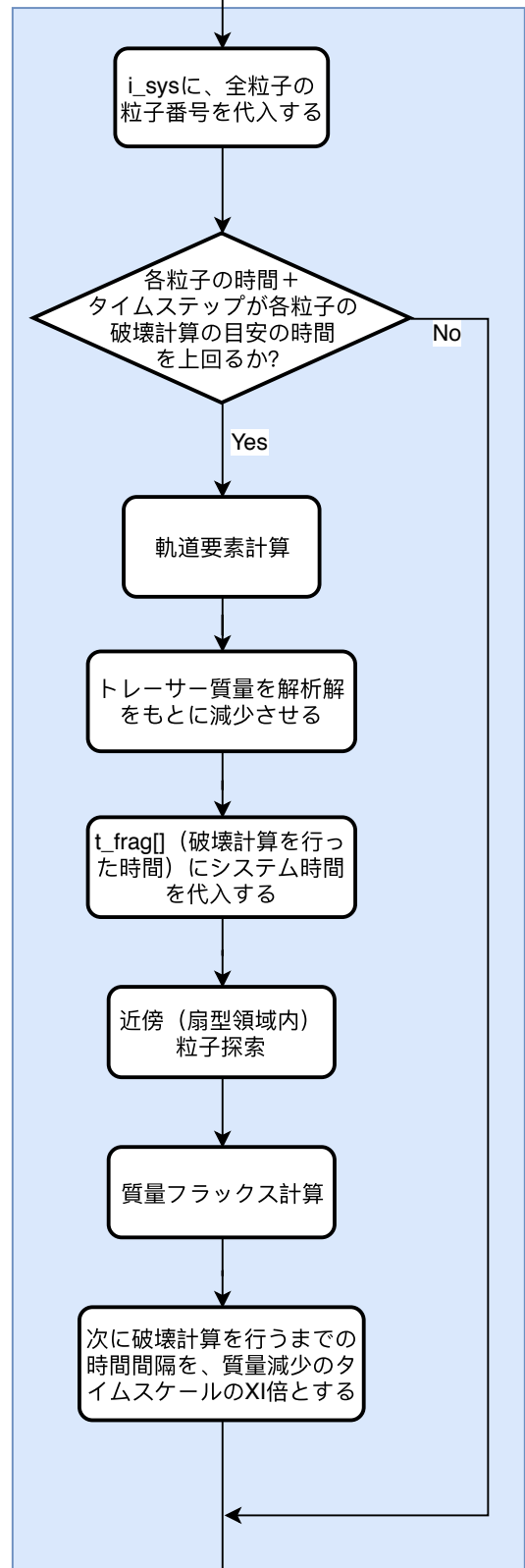
全ての粒子の修正子
を計算する

全ての粒子の修正子を
2回イテレーション

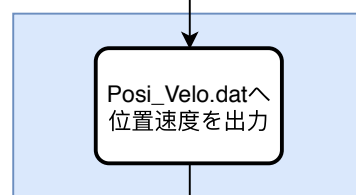
#if FRAGMENTATION



#if FRAGMENTATION



#if POSI_VELO_FILE



#if ENERGY_FILE

中心星を含めた
重心計算

全エネルギー計算

全角運動量の大き
さ計算

Energy.datへエネルギー
と角運動量を出力

#if ORBITALELEMENTS_FILE

全粒子の軌道要素計算

Planet%02d.datまたは
tracer%06d.datへ軌道要素を出力

離心率と軌道傾斜角の
二乗平均平方根を計算

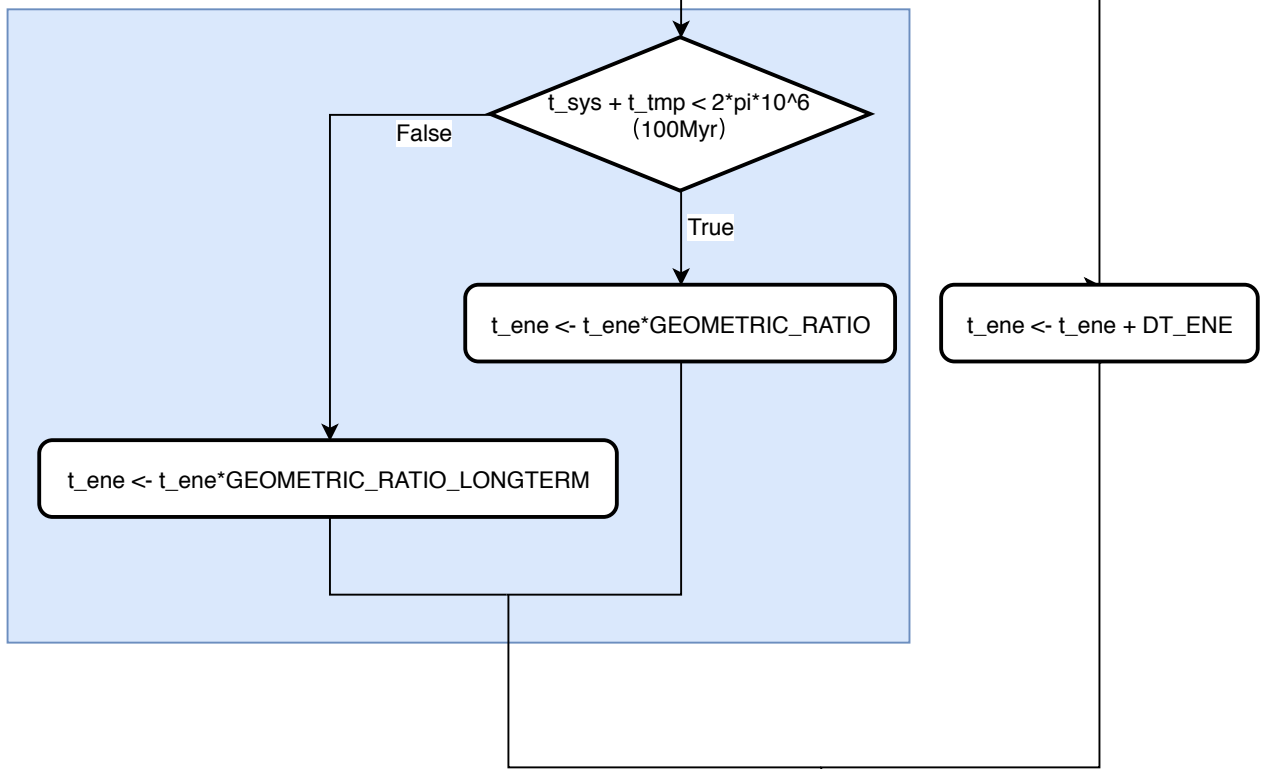
#if DT_LOG

tempread.datへ位置、速度、
加速度、質量、タイムステッ
プなどのデータを出力

#if FRAGMENTATION

tempfragread.datへ
破壊計算に使った
データを出力

#if DT_LOG



Decision: $t_{\text{sys}} + t_{\text{tmp}} > t_{\text{fragcheck}}$

False

True

基準とする惑星の軌道長半径から
+/-0.5*DELTA_HILL*
(相互ヒル半径)の範囲にいる粒子の総
数をcenterとする
それより内側にいる粒子の総数をinner、
外側にいる粒子の総数をouterとする

#if TRACERLIST_FILE

tracerlistnumber.dat
^inner, center,
outerの数を出力

#if FRAGMENTATION

inner, center, outerの
範囲の総質量と面密度
を計算

Sigma_dep.datへ質量や
面密度のデータを出力

Posi_Mass_%03d.datへ位置
や質量のデータを出力

$t_fragcheck \leftarrow t_fragcheck * GEOMETRIC_RATIO_FRAG$

太陽半径以内に入った、
または系外へ出て行った

No

Yes

#if ELIMINATE_PARTICLE

粒子を消す
空いた粒子番号には、最後
尾の粒子を持ってくる

このステップで衝突が起きたか?

No

Yes

全ての粒子の時間をシステム時間に合
わせる
各粒子のタイムステップはとりあえず
非常に小さな2進数とする(中途半端
な時間で衝突が起きている場合、全粒
子の時間がそろわなくなるため)

各粒子の時間+タイムステップが一番小
さい粒子集団について、次のタイムス
テップの候補を計算し2進数に合わせ、位
置、速度、加速度、加加速度を更新する



ヒープソートを用いタイムステップが
小さな順に粒子を並び替え
index[]に並び替えた粒子番号を格納

タイムステップが一番
小さい粒子集団の総数
をn_i_sysとする

タイムステップが一番
小さい粒子集団のうち
一番若い粒子番号を
i_sysとする

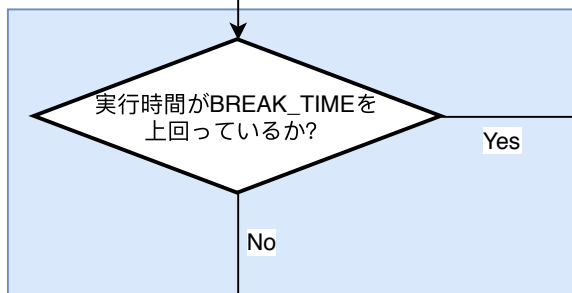


No

Yes

log.datへステップ数、粒子数、
システム粒子、タイムステッ
プ、システム時間などを出力

#if EXECUTION_TIME



Yes

No

