# GRAPE-DR 移行マニュアル

## 1 GRAPE-6 モード

GRAPE-6 関数では完全に互換性が保たれているわけではありません。利用可能な最大 j 粒子数は約 260 万体です。i 粒子のパイプラインは 256 です。以下が主な互換性情報です。

### 1.1 近傍粒子リストは実装されていない

GRAPE-6 で利用可能であった近傍粒子リスト機能は、GRAPE-DR では今のところ実装されていません。

### 1.2 cluserid を指定しない関数の追加

GRAPE-DR では clusterid を指定する必要の無い関数が追加されました。これらを使えば、全モジュールを使って計算する場合は、明示的に clusterid を指示する必要がありません。関数名は、末尾に"_all"がつき、clusterid を指定する引数が省略されます。以下、g6_*_all 関数の一覧です。これらに対応する関数をご使用の場合は、書き換えて下さい。

- void g6_open_all(void);
- void g6_close_all(void);
- int g6_set_j_particle_all(int address, int index, double tj, double dtj, double mass, double a2by18[3], double a1by6[3], double aby2[3], double v[3], double x[3]);
- int g6_set_j_particle_mxonly_all(int address, int index, double mass, double x[3]);
- void g6_set_ti_all(double ti);
- void g6calc_firsthalf_all(int nj, int ni, int index[], double xi[][3], double vi[][3], double fold[][3], double jold[][3], double phiold[], double eps2, double h2[]);
- int g6calc_lasthalf_all(int nj, int ni, int index[], double xi[][3], double vi[][3], double eps2, double h2[], double acc[][3], double jerk[][3], double pot[]);
- int g6calc_lasthalf2_all(int nj, int ni, int index[], double xi[][3], double vi[][3], double eps2, double h2[], double acc[][3], double jerk[][3], double pot[], int nnbindex[]);
- int g6_read_neighbour_list_all(void);
- int g6_get_neighbour_list_all(int ipipe, int maxlength, int *nblen, int nbl[]);
- void g6_set_nip_all(int nip);
- void g6_set_njp_all(int njp);
- void g6_set_i_particle_scales_from_real_value_all(int address, double acc[3], double jerk[3], double

phi, double jfactor, double ffactor);

- void g6_set_i_particle_all(int address, int index, double x[3], double v[3], double eps2, double h2);
- int g6_get_force_all(double acc[][3], double jerk[][3], double phi[], int flag[]);

### 1.3 移行手順

前述の互換性情報をもとに以下を確認して下さい。

1. 関数名_all の対応する関数に書き換え、idcluster を外す。
2. いくつか削除された関数がある。計算に必要な関数か確認する。

## 2 GRAPE-5 モード

### 2.1 追加機能

GRAPE-5 モードに最近接粒子探査に関する機能が追加されました。以下の関数です。

#### 2.1.1 高レベルインターフェイス

■void g5n_set_jp(int adr, int nj, double *m, double (*x)[3], int *index); 引数 index には、j 粒子にインデックス値を割り当てます。見つけた最近接粒子は、このインデックス値が返されます。

■void g5_set_index(int ni, int *index); i 粒子インデックスを設定し、i 粒子と j 粒子で同一のものを最近接粒子として判定しないようにします。

■void g5n_get_force(int ni, double (*a)[3], double *pot, double * rnnb2, int * innb); 加速度を取得します。その際に、最近接粒子までの距離の二乗 (rnnb2) とインデックス値 (innb) を返します。

■void g5n_calculate_force_on_x(double (*x)[3], int * index, double (*a)[3], double *p, double * rnnb2, int * innb, int ni); g5_set_xi, g5_set_index, g5_run, g5_get_force などを呼んで、加速度を計算し返します。index は i 粒子のインデックス値です。最近接粒子までの距離の二乗 (rnnb2) とインデックス値 (innb) を返します。

#### 2.1.2 低レベルインターフェイス

- void g5n_set_jpMC(int devid, int adr, int nj, double *m, double (*x)[3], int *index);
- void g5_set_indexMC(int devid, int ni, int *index);
- void g5n_get_forceMC(int devid, int ni, double (*a)[3], double *pot, double * rnnb2, int * innb);
- void g5n_calculate_force_on_xMC(int devid, double (*x)[3], int * index, double (*a)[3], double *p, double * rnnb2, int * innb, int ni);