mdspass2改良点

2022年01月16日

* Win版ver.20181222（講義で使用）に，CNTの構造生成を繰り返すと落ちるというバグが判明．create\_config.cpp中で，atop.asp[iatom]=atom.asp[j]と書いていたことが原因．strcpyを使った書き方に修正した．
* 同Win版で，Create config 🡪 Edit atom 時にCreate config のcloseボタンがdisableになっておらず（クリックできる状態のままになる），クラッシュの元となるので修正した．

(mdspass2.20181222mとしてアップした)

2018年10月21日

captureで不具合が出ることが判明（Linuxでは黒塗り画像になる、Windowsでは原子の球が真っ白で表示される・窓をリサイズした後にキャプチャすると化ける）したため、修正。capture.cpp内のRGBAをRGBに変更（2か所）、glPixelStoreiをGL\_PACK\_ALIGNMENTに対しても1にセットする(GL\_UNPACK\_ALIGNMENTも1に)ことで修正できた模様。

2016年7月14日

* bookkeepingに存在したバグを修正。orthogonalなセルでないとき（セル形状ベクトルが直行していない場合）に不具合が出ていたものを修正。
* bookkeepingを行う際、orthogonalとnon-orthogonalに場合分けし、後者の場合は規格化された原子座標qx, qy, qzを用いて最近接レプリカ原子を探索するアルゴリズムに変更した。これに伴い、qx, qy, qzを保持する配列を用意した。
* 現状のセットアップ状況をSETDATに描きだせる機能を追加。

2016年6月22日

* NEB計算において、原子が変位する最大距離を見積もり、自動的にブックキーピングし直すようにした。従来はユーザーがあらかじめ大きめのカットオフ距離を設定してやる必要があり、それが足りないとエネルギーカーブが正しく得られない（不連続点ができる）という問題があったが、これを解消した。

2016年6月21日

主にメモリリーク関係のバグをつぶしました。

* delete[] x, y, z 🡪 delete[] x; delete[] y; delete[] z と書かなければいけなかったので、この点を修正。（対象ファイル多数）ｓ
* array\_set.cpp の中を、new と delete 文に統一(malloc,free等の使用を止めた)。ただしメモリリークはこれが原因ではなかった。
* e\_force\_mishin.cpp 内で delete[] の所のバグを修正（typoのため配列が解放されていなかった）。
* e\_force\_mishin.cpp 内で、メッシュデータを呼ぶときに引数がデータ範囲外となった場合にセグフォで落ちる現象があったので、データ範囲を強制的にmax/min内に収めるようにした。NEB等で構造が変になった時に起こる現象。

mdspass2 改良点について（2016年6月17日）

1. mdspass2が時間がたつと落ちてしまう不具合を修正

前回、原子間ボンドを立体的に表示する機能を追加した際、配列処理の不具合によりメモリリークを起こしていたことが原因と判明。確保したメモリ領域を処理後に解放する命令が抜けていたので、これを修正した。

🡪挙動は安定した模様であり、解決したと思われる。

1. 表面積計算不具合を修正

前回、CNTの断面形状が真円でない場合にも表面積が計算できるようにルーチンを変更したが、断面の中心から最外殻層の原子位置をr(**)のように極座標表示してその経路を計算していた。ところが断面がダンベル型のような形状になった場合、r(**)が二価関数となってしまうことがあり、その場合には正しく経路が計算できないことが判明した。

そこで、最外殻のある原子を出発点として、その原子と結合している原子を次々に検索していくことで、最外殻の形状を求めるアルゴリズムに変更した。

🡪今度こそ正しく計算されていると思われるが、注意してチェックしてほしい。

1. 構造緩和計算に”re-heat”機能を追加

ある一定圧力のもとで構造緩和計算（座屈／非座屈判定）を行う際、Fmaxがある値(tolfor)以下になるとMDを停止する、という方法を取るのが一般的であるが、不安定平衡状態（座屈荷重に達しており系が不安定状態になっているにも関わらず、荷重が釣り合ってしまうために変形が起こらない）に陥ってしまうことがある。不安定平衡状態に陥っているかどうか判定するためには、その状態で少しだけ温度を与えてMD計算を行い、原子を「少し揺らしてやる」という方法が考えられる（再加熱：re-heatと称する）。

そこで、Fmax<tolforとなった後、MDを停止するのではなく、一定温度のMD（NVTアンサンブル）を一定ステップ実行し(re-heatプロセス)、その後再び緩和計算に入る、という機能を追加した。

Set parameter パネル内の”Stop MD by Fmax”の下に、”Reheat after stop”チェックボックスを設けた。両方にチェックが入っている場合にこの機能は有効となる。その少し下にある”Reheat param”をロールアウトすると、詳細設定ができる。re-heatプロセスにおける温度、MDの時間ステップ、ステップ数が設定できる。”# of reheats to go”はこのre-heat→緩和を何回行うかを指定するものである。例えば2にしておくと、緩和→re-heat→緩和→re-heat→緩和、となる（窓に表示される値はre-heatが行われる毎にカウントダウンされる）。

1. 多層CNTの特定の層だけを表示する機能を追加

“Set parameters”パネルの下の方にある”Special settings for CNT”ロールアウトの中に、”Which layer to show”という項目を追加した。Allにチェックが入っているとすべての原子を描画するが、このチェックが外され、1st, 2nd, ... のチェックボックスがONになると、多層CNTの層ごとに描画を行うことができる。1stは最外殻層、2nd はそれより一つ内側の層、・・・である。

　ただしこの機能は、各層がz軸にほぼ平行（xy断面形状がどこでも同じ）になっていることを想定している。Corrugationに対しては問題ないはずであるが、軸方向圧縮などにより座屈が起こっている場合には対応できない。

mdspass2 改良点について（2016年5月19日）

1. 原子間ボンド表示について

* Set param 🡪 Setup for drawing とクリックすると開くボンド表示設定部分(“Bond”と書かれた枠で囲まれた部分)に機能を追加しました。
* ボンド表示のタイプをLine/Cylinderから選べるようにしました。前者は立体感の無い線で、後者は立体感のある棒で表示します。描画速度は当然後者の方が遅くなりますが原子数が少ない場合には問題ないでしょう。
* Cylinderの場合、色が選択できるようにしてあります。下向き矢印の付いたウインドウを選ぶとリストが表示されます。
* ボンドの太さを変えることができるようにしました。Lineの場合、7で頭打ちとなりそれ以上の値を入れても変わりません。
* なお、この場合のボンドは、Bond lengthで表示されている値（単位はÅ）範囲内にある原子同士を単純につなげているだけです。結合状態を計算しているわけではありません（古典分子動力学ですので）。

1. 特定原子のマーキングについて

* Set param 🡪 Atom color で、原子の着色法を設定するパネルが開きます。ここで、特定の原子3つまでを選択し、好みの色で表示するという機能を付加しました。
* Marked atom 0/1/2 で原子の番号を選択します。原子の番号が分からない場合には、MD viewerウィンドウ上で原子をクリックすれば、コンソール画面上（Windowsではコマンドプロンプトのような黒いウィンドウ）に選択した原子の情報が表示されるので、それを参考にしてください。着色を止めたければ、この値をゼロにしてください。
* 選択した原子にそれぞれ別の色を与えることができます。

1. 圧力の算出法について

* ナノチューブの解析の場合（Set param 🡪 Special settings for CNT 🡪 Corrugation modeをチェックした場合）、Stressでパネルを開くとCNT corrugationの部分に圧力が表示されるようになっていました。この時チューブの最外層の表面積を求めるルーチンが、断面形状＝真円を仮定していたため、断面がつぶれた時に間違った値を吐く、という問題がありました。
* そこで、今回は断面がつぶれた場合にも正しく表面積を出してくれるように改良しました。多層ナノチューブの場合、最外層のチューブのみを検出してその表面積を計算してくれるようにしています。ユーザー側で特に行うべき操作はありません。
* これに伴って、Outermost radiusの表示を廃止しました。

ナノチューブ表面積計算ルーチン（20160519版）

* 1. チューブの中心がｚ軸に沿うと仮定する。中心軸の位置(x,y)を決め、中心軸から極座標表示した原子の座標r()を計算し、メッシュデータとして記憶する。  
     , rをそれぞれ the[i], rad[i] と配列変数に保存する。iはカウンタで、要素の数はnum\_elemとする。この時点ではnum\_elem = 原子数。
  2. the[i], rad[i] をthe () の昇順にソートする。
  3. 最外殻のみを取り出す。iを0から一つずつ増加させていき、要素iとその次の要素に対する半径rの差drad = rad[i] - rad[i-1] を求める。この値が負（かつ絶対値が閾値より大）であれば、最外殻でない原子が見つかったことになるので、その要素を削除する（それより大きな要素に対するデータを一つずつ下げる）。値が正（かつ絶対値が閾値より大）であれば、要素iおよびそれより小さな要素はすべて最外殻でない原子のデータであるので、これを削除する。
  4. これで最外殻の原子の座標r()が  の昇順に得られた。密に詰まりすぎている要素を削除しておく。
  5. 得られたr()を用いて、を計算し、セルのｚ方向長さをかけるとチューブの表面積が得られる。