

生体分子のための対話型分子グラフィックスシステム LIVE の開発†

石井 隆 寛** 池口 満 徳**
山田 俊 宏** 土井 淳 多**

蛋白質や核酸などの大きな生体分子の構造を多角的に理解しモデリングするための対話型分子グラフィックスシステム LIVE を開発した。使いやすいユーザインタフェースを持ち、分子構造を手軽に定量的に表示できるシステムをめざした。表示形態は、線画以外に、ドットサーフェイスモデル、空間充填モデル、ボール&スティックモデル、リボンモデル、円筒モデルとこれらの半透明表示であり、陰影付けされたリアルなイメージの生成ができる。これらの表示形態と色の設定は論理演算によって定義・登録された原子のグループに対して行うので、分子を原子ごとに任意の形態や色で表示できる。ウィンドウシステム上にユーザインタフェースを構築しほとんどの操作はマウスで行う。分子の回転・移動操作もマウスで2軸同時に扱う。常に分子の現在位置や方向を数値で表したり、俯瞰図の表示をリアルタイムに行う機能により表示状況の把握を容易にする。マウス操作は、パッチとして履歴を残せる。パッチ処理は処理の制御や変数の使用も可能である。原理的に取り扱う分子の大きさや数に制限がなく、分子の編集や分子データの入出力が高速になるように設計した。本システムにより生体分子のシミュレーションの研究に貢献できるものと期待している。

1. はじめに

分子生物学においては蛋白質や核酸等の生体分子の機能やそのプロセスを知るために、実験はもとよりそれらの構造をよく理解し物理的・化学的特性を考察することが重要である。さらに、蛋白質工学の分野では、分子生物学で得られた知見をもとに未知の分子の構造や機能を予測して新しい機能を持った生体分子の設計をめざしている。そのため、多くの分子設計システムが開発されてきた^{1)~8)}。これらのシステムはX線結晶解析や核磁気共鳴(NMR)による構造解析から得られた原子の座標データをもとに、3次元的な分子の表示、編集をはじめ分子軌道法、分子力学⁹⁾、分子動力学^{10), 11)}による静的力学計算、動的力学計算、および統計解析等の数値計算を行うもので、いわゆる化学CADである。これらのシステムのなかでパーソナルコンピュータ上で実現されたものは、手軽に操作できる反面、取り扱える原子の数に制限があったり処理速度に問題があり、高分子を取り扱うのが困難である。一方、ワークステーション(WS)以上の計算機上のシステムでは、機能が豊富ではあるが操作が複雑で、かなり特殊な知識と長期間の訓練が要求される。そこ

で、操作が簡便でしかも3次元の分子構造の表示・編集機能が豊富な、すなわち、分子構造を手軽に定量的に表示できる分子グラフィックスシステムをめざしてLIVEを開発した。LIVEはWS上のシステムで、分子表示・編集に関しては他の分子設計システムとほぼ同様の機能を持つ。そして、必要に応じて前述のような数値計算を付加したり他の設計手法を柔軟に組み込めることをめざした。本論文では述べないが、現在、リガンド-受容体のドッキングの自動化とアミノ酸1次配列から蛋白質の2次および3次構造予測を行うソフトウェア等が組み込まれている。

LIVEの特徴は次のようになる。

(1) 将来の拡張のために、ユーザインタフェースを描画プログラムとは独立にし、必要があればこれを容易に別のものと交換可能にした。

(2) マウスとキーボードを入力装置とし、操作を単純化するとともにダイアル、ボタンスイッチなどの装置依存性を少なくした。作業の進行はマウスによるポップアップメニューの選択であるが、特に頻繁に行う分子の回転・移動はマウスを用いて2軸同時に操作できる。このため、コマンドや他の装置の扱い方をユーザが覚える負担を軽くし、効率を上げることができ。

(3) 1分子内の原子を論理演算を用いて様々な条件でグループ化し、そのグループに対して表示形態や色の設定を行える。このため、注目したい部位を強調して表示したり、分子内部の構造を容易に明確にする

† Development of an Interactive Biomolecular Graphics System LIVE by TAKAHIRO ISHII, MITSUNORI IKEGUCHI, TOSHIHIRO YAMADA and JUNTA DOI (Department of Biotechnology, Faculty of Agriculture, The University of Tokyo).

** 東京大学農学部応用生命工学専攻

ことができる。

(4) 奥行感のあるリアルな分子イメージを様々な表示形態で表示できるよう、半透明表示も含め多種の陰影付け表示を用意した。これに、(3)の特徴を合わせて用いれば、簡単な操作でさらに多様な表示が可能になる。

(5) 表示中の分子の姿勢や位置を常時、数値で表示し、表示画面が分子イメージのみならず並進・回転の6自由度に対し定量性を持たせるようにした。

(6) 分子の多画面表示機能の一つとして、俯瞰図の表示をリアルタイムに行う。これで画面に投影されている3次元的な表示領域、分子間の関係、分子の外形などを大局的にとらえることができる。

(7) バッチ処理に、条件分岐・ループなどの制御、変数の使用、値の四則演算・論理演算等の機能を持たせ、複雑なアニメーションの作成を可能にした。また、マウス操作は履歴を取れるので、後に操作を再実行したり、バッチを記述するときの基本書式にすることができる。

2. システムの構成

2.1 システムの概要

LIVE の機能を表すものとして、主なコマンドを表1に示す。これらは分子データの入出力、表示、編集、原子のグループ分けに関するものでキーボード入力とバッチファイル記述に使われる。これに対応する操作はマウスで可能であり、実用的にはマウスを用いたポップアップメニューの選択により作業を進める。

システムの起動は1分以内に完了し、実行時にはユーザインタフェース部 uif と描画部 dcore の二つのプロセスが同時に走る。さらに、分子データを変換するためのプログラム pdbcvt を必要に応じて起動する。

動的に配列の大きさを変えられ柔軟なデータ構造を取れるC言語で記述してあり、原理的に扱える原子数や分子数の制限はない。また、コマンド入力の字句解析や構文解析部を設計・開発する際にもC言語を採用した。

蛋白質の3次元構造データとしては特によく用いられる Protein Data Bank (PDB)¹²⁾ を分子構造データとして使うが、他の書式のデータベースに対しても柔軟な対応を可能としている。

なお、LIVE は UNIX ワークステーション、HP 9000 モデル 350 SRX 上の HP Windows/9000 環境

表1 主な LIVE コマンド
Table 1 Main LIVE commands.

コマンド	機能
load	分子ファイル読み込み
save	分子ファイル保存
quit	live の終了
purge	分子の削除
rot	分子の回転
trans	分子の移動
rot.abs	オイラー角による回転
trans.abs	絶対座標による移動
reset	ロード時の位置に戻す
local	分子群の設定
global	グローバルモードにする
ch.center	回転中心の変更
zoom	画面の拡大・縮小
window	表示領域の設定
mols	ロード分子一覧
ren	分子名の変更
clip	画面の前後面の設定
distance	視点から原点までの距離
depthcue	デプスクューを設定
perspect	遠近画法に設定
parallel	平行投影に設定
setg	原子グループの登録
freeg	原子グループの登録破棄
form	表示形態の変更
color	表示色の変更
selatom	マウスで原子選択
cut	原子の結合の切断
connect	原子の結合の生成
part	原子の分割
merge	分子の一体化
copy	分子のコピー
modify	分子構造の修正
log	操作の保存
batch	バッチコマンドの実行
print	変数の内容表示
cd	ディレクトリの変更

で開発した。

2.2 システムの構造

図1にシステムの構造を示す。一貫性のあるアーキテクチャ、比較的簡単に複数の入力を同時に扱う処理の実現、開発時の分業のしやすさ、将来への拡張を考慮して、ユーザインタフェース部 uif と描画部 dcore を独立させた。これは、uif をそっくり変更することで特定用途向きのシステムが構築できることや特定ユーザ向けのシステムが容易に作れることを目的としたためである。

uif はウインドウシステム上で実行されるインタ

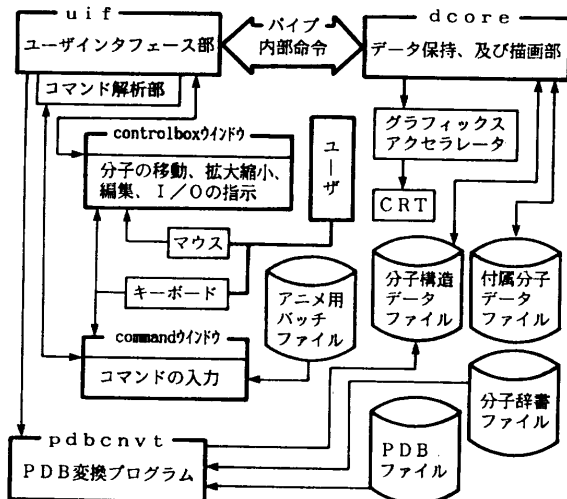


図 1 システムの構造
Fig. 1 System configuration.

リタである。マウス入力やコマンド入力を解釈して機械語命令に類する内部命令に展開したりユーザへ入力を促したりする。また dcore は uif から送られてくる内部命令に基づいて、分子の絵をグラフィックスアクセラレータを通して描画するプロセスである。そのため dcore は分子構造データファイルおよび付属分子データファイルの読み込みと書き込みも受け持ち、分子表示に必要なデータを保持する。これら二つのプロセス間の通信には双方向パイプを使い内部命令を受け渡す。

ユーザは uif が管理する controlbox ウィンドウ (図 6 の右側部ブルーの部分) に対してマウスで分子の移動、拡大縮小、表示形態、編集、および分子データの入出力等を指示する。xy 座標 (図 6 で緑のカーソルが示す部分) に対して回転や平行移動を一度に指示できるのが特長である。コマンドやファイル名や数値の入力には command ウィンドウを用いる。

分子の読み込みを例にして命令の流れを図 2 に示す。内部命令は整数型の命令番号とそれに続く複数のデータで一つの命令を成して、描画やファイル入出力や編集等に必要な最小限度の命令セットである。dcore はこれを標準入力から受け取り実行したあと応答に必要なデータを標準出力に出力する。uif は、出力パイプを dcore の標準入力に、また入力パイプを dcore の標準出力に結合させており、マウス入力をその操作に必要な複数の内部命令に展開した後、次々と出力パイプに送り込んで応答を入力パイプから読みだす。ユーザやバッチファイルからコマンドが入力され

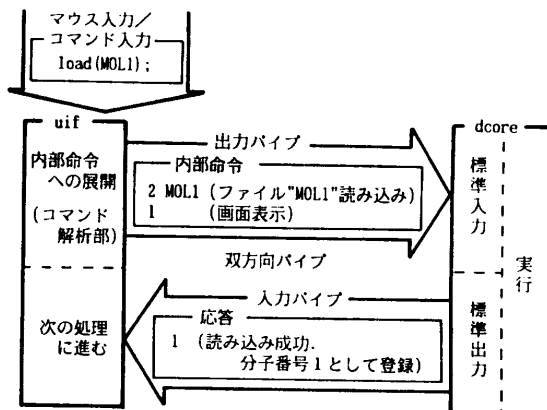


図 2 uif と dcore 間の通信
Fig. 2 Communication between uif and dcore.

た場合はコマンド解析ルーチンを通して構文解析されたあと内部命令列に変換される。構文規則の仕様を与えると LALR (1) 構文解析法に基づくパーザの C 言語ソースコードを出力する yacc¹³⁾ を用いてコマンド解析ルーチンの開発を行った。

分子構造データファイルは、分子表示に必要な構造データを記述したバイナリファイルであり、次節で述べるように PDB を変換して作成される。これだけで基本的な分子構造を表すことができるが、1分子全体に関する情報、例えば分子の名前、画面に対する位置・姿勢、回転中心と表示形態の設定等はエディタで確認したり修正できるよう別にテキスト形式の付属分子データファイルとして独立させた。

2.3 PDB の LIVE 用データへの変換

PDB の中には原子座標や結合情報が不足していたり、特殊な残基が含まれていることがある。また、図 3 のように分子-残基 (アミノ酸)-原子といった階層的な構造データ以外に各原子や残基の色、表示形態、および原子のグループ (後述) に関する情報をも蓄えておく必要がある。そこで、PDB を直接分子データ

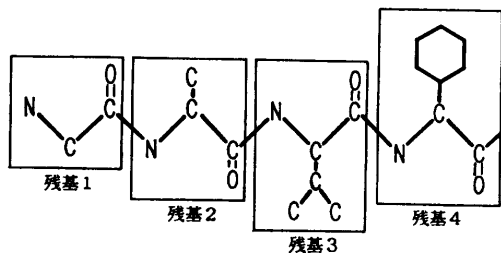


図 3 分子の構造
Fig. 3 Molecular structure.

として扱わず、読み書きが速くコンパクトなバイナリファイルに変換して分子構造データファイルとすることが望まれる。そのためのプログラムが `pdbenvt` である。

`pdbenvt` は、含まれる原子の数や名前、原子間共有結合情報等を残基別に記述した辞書ファイルを参照しながら、(1)読み込んだ PDB の原子間結合関係をチェックし、(2)原子間結合関係と原子の名前と残基の名前から原子の種類、残基の種類を分類し、(3)結合情報を補い、(4)辞書に未登録の残基や、登録されていても原子間結合関係の異なる残基があれば新しく登録し、(5)バイナリファイルを出力する。こうして出力された分子構造データファイルは、原子座標とともにすべての共有結合と水素結合の情報をも含み、それ自身で一つの分子表示に必要と考えられる全データを網羅している。それにもかかわらず、原子や残基は同じ名前や属性を共有することによって、PDB に対して平均 33% の大きさになり大幅なファイル圧縮が可能となった。

2.4 描画部 `dcore` 内部の分子データ構造

分子構造データファイルは `dcore` によって入力される。`dcore` 内部で保持される分子のデータ構造を示すのが図 4 である。この図で `atm` は、原子の座標・種類・色・フラグ・共有結合でつながる相手原子の `atm` ポインタの配列等を持つ構造体であり、つながりリストの要素である。さらに `res` もつながりリスト構造をとる構造体で、残基の種類・色・フラグ・`atm` リストの先頭ポインタ等を持つ。また `mol` 構造体は分子の名前・色・`res` リストの先頭ポインタ等を持つ。このようなデータ構造にした第 1 の理由は、すべての `atm`

に結合すべき相手を覚えさせ、あらゆる分子に対して柔軟なデータ構造にするためである。例えば、同じ名前であるにもかかわらず二つの残基が、異種の原子を含んでいたり原子数や原子間結合関係が異なっている場合であっても分子構造の表現は可能となる。第 2 の理由として、つながりリスト構造は分子間または分子内の編集が高速に実行できるためである。

2.5 原子のグループ化

分子内の部分的な構造を見たいときや強調したいときがある。そこで、多くの原子の中から設定された条件に合致したものをグループ (Conditioned Atom Group; CAG) としてまとめ、これに対して一度に表示形態や色の設定をすることができる。この条件は基本的な原子選択条件の論理演算として表現する。指定された CAG は 1 分子につき 16 個まで登録することもでき、後で再表示や別の CAG との論理演算にも使用できる。基本的な原子選択条件として指定できるのは、(1)原子名や残基名の正規表現、(2)残基の通し番号の範囲、(3)原子の通し番号の範囲、(4)現在表示中の原子や残基の表示形態、(5)表示中の原子や残基の色、(6)主鎖または側鎖、(7)球および直方体領域 (この領域内に含まれる原子群を選択)、(8)共有結合する 2 原子 (原子の共有結合ネットワークにおいてこの原子間の結合を境に 2 分割される領域の片方に属する原子群を選択)、(9)蛋白質の 2 次構造 (ヘリックス、シート、ターン、のいずれか)、(10)CAG 番号、である。

例えば、「分子 7 に含まれるグリシン残基内の原子のうち、CAG 番号 12 のグループにも属さないし炭素原子でもない原子の色を赤にせよ。」は、

```
color(%7, red, res[GLY]&! (cag[12]|atom[C*]))
```

となる。ここで、`!` は否定、`&` は論理積、`|` は論理和を表す演算子、`*` は 0 文字以上の任意の文字列を表す正規表現である。また 12 番の CAG はすでに登録されているものとする。

CAG は、前節で述べた `atm` 構造体のメンバであるフラグ (2 バイト=16 ビットの変数) の各ビットを使って実現される。例えば、CAG 番号 3 のグループに属する原子 (`atm` 構造体) はフラグの最下位から 3 番目のビットが 1 になっている。`!`、`&`、`|` の演算は、二つの CAG 間の論理和、論理積、CAG の反転を行う内部命令を、`uif` がスタックを用いて構文木どおりに生成することで実現した。

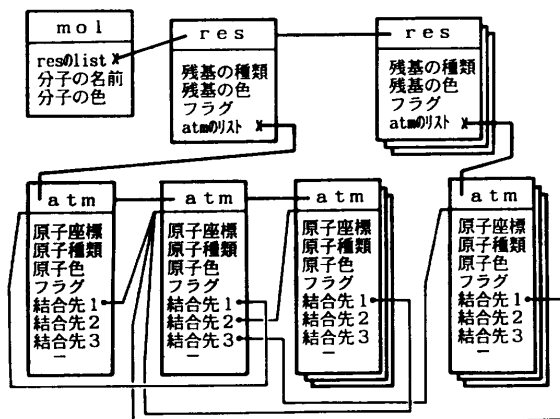


図 4 分子のデータ構造

Fig. 4 Molecular data structure.

2.6 分子の表示形態と配色

分子構造をより良く理解するためには、陰影付けされたソリッドモデルの表示がもっとも望ましい。そこで、分子を線画やドットサーフェイスだけでなく、陰影付けされた空間充填モデル（ファンデルワールス半径球モデル、半径は可変）、ボール & スティックモデルで表現し、原子や残基の名前も3次元的に表示できるようにした。特に蛋白質に関して2次構造の理解を一層進めるために主鎖のリボン表示、ヘリックス部の円筒表示、およびシート部の曲面表示も用意した。さらに空間充填モデル、ヘリックス部の円筒、リボンは半透明な表示も可能である。

これらのどの表示形態でも原子または残基の種類に従って色分けができる。また、特に蛋白質の主鎖のリボン表示ではヘリックス、シートおよびターンの色分けも用意した。

通常、表示形態と色は原子すべてに画一的に指定するが、CAGと組み合わせれば1分子を任意に表現できる。上に述べた表示形態を1原子に関して重複して表示したり、部分的に表示形態を変えたり、部分的に配色することができる。

円筒と球は多面体で近似し Gourand シェーディング¹⁴⁾を行って表示する。また、リボンは双2次のBスプライン曲面のパッチに同じく Gourand シェーディングを行って表現する。

2.7 分子の移動操作

分子の姿勢や位置を変える移動操作はもっとも頻繁に行われかつ重要である。しかし、これを従来のようにダイアルで行うことは必ずしも操作性が良いとはいえない。すなわち、ダイアル使用時に視線がディスプレイから離れてしまい思考が中断したり、通常6個のダイアルを扱うことが必要で訓練を要するからである。そこで、マウスでx軸とy軸を同時に扱う回転と平行移動を用意した。これによりあたかも回転中心を固定して指でその分子を押したような感覚の回転や指でつまんで動かすような感覚の平行移動ができ、x軸y軸を余り意識せずに操作できる。ただし、画面に対して水平右向きをx軸、垂直上向きをy軸、ユーザに向かう向きをz軸とし、また回転中心は表示領域の中心または個々の分子が持っている基準点である。これらの操作はどのような表示をしているときでも可能であり、個々の分子に対しても（ローカルモード）、全分子に対しても（グローバルモード）行うことができる。後者は、視点の移動に相当する。なお、マウスで

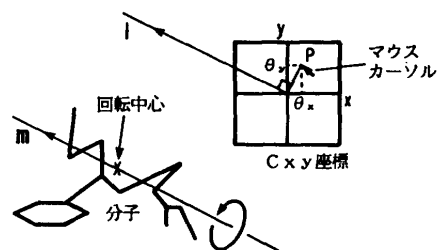


図5 Cxy座標を用いた回転操作
Fig. 5 Rotation with an icon in a window for control.

ピックした原子や数値で与えた座標に分子の基準点を変えることもできる。

回転モードにし controlbox ウィンドウ内の xy 座標 (Cxy 座標) をピックすると、図5に示すように、直線 m (直線 l と画面に平行で回転中心を通る直線) を軸として分子は相対的に回転する。Cxy 座標の原点から遠い位置をピックすれば回転速度が上がる。また、平行移動モードにし Cxy 座標上でドラッグ (マウスボタンを押したままカーソルを移動) すればカーソルの動きにしたがって xy 平面に平行に分子は移動する。ただし、z軸回りの回転とz方向の移動は別に設けた Cz 座標 (図6で赤のカーソルが示す部分) を使う。

ここで、回転操作に使った座標変換行列 R は、Cxy 座標で指示する点を $P(\theta_x(\text{rad}), \theta_y(\text{rad}))$ 、回転中心を3次元座標原点、さらに

$$\theta = (\theta_x^2 + \theta_y^2)^{1/2}, \quad nx = -\theta_y/\theta, \quad ny = \theta_x/\theta$$

とすると、

$$R = \begin{bmatrix} nx^2 \cdot (1 - \cos \theta) + \cos \theta & & & \\ nx \cdot ny \cdot (1 - \cos \theta) & & & \\ ny \cdot \sin \theta & & & \\ & nx \cdot ny \cdot (1 - \cos \theta) & -ny \cdot \sin \theta & \\ & ny^2 \cdot (1 - \cos \theta) + \cos \theta & nx \cdot \sin \theta & \\ & ny \cdot \sin \theta & \cos \theta & \end{bmatrix}$$

と表せる。ただし、 $(x', y', z') = (x, y, z)R$ で変換を行う。ここで、 θ_x, θ_y は、オイラー角ではなく、軸 $(nx, ny, 0)$ を中心に右ねじ回りに θ (rad) 回転させる意味における角度であり便宜的な定義であるが、どちらかが零の時は正確にx軸またはy軸回りの回転角度に一致する。

以上の方法は、ダイアルを使わない効率的な操作として一般の3Dグラフィックスにも適用できる。

2.8 分子の位置と姿勢の定量化

定量的に分子の位置と姿勢を把握するために、指定

した分子に対してこれらを数値で常時表示するモードを設けた。位置として、分子表示領域の中心を原点として分子の基準点の3次元座標をÅ単位で表示する。姿勢としては、 x 軸回りに a (deg), y 軸回りに b (deg), z 軸回りに c (deg) の順番で右ねじ回りに回したときの三つのオイラー角 a , b , c の値を表示する。

表示分子について 3×3 の回転座標変換行列を M , また a , b , c を変数とする関数が要素となる3軸回りのオイラー回転の行列を Y とすると,

$$ca = \cos a, \quad cb = \cos b, \quad cc = \cos c \\ sa = \sin a, \quad sb = \sin b, \quad sc = \sin c$$

等として,

$$Y = \begin{bmatrix} cc \cdot cb & & & & & \\ -ca \cdot sc - cc \cdot sb \cdot sa & & & & & \\ sa \cdot sc - ca \cdot cc \cdot sb & & & & & \\ & cb \cdot s & & sb & & \\ & ca \cdot cc - sa \cdot sb \cdot sc & & cb \cdot sa & & \\ -cc \cdot sa - ca \cdot sb \cdot sc & & ca \cdot cb & & & \end{bmatrix}$$

と表せるので, $M=Y$ とおいて9個の独立でない連立方程式を解けば a , b , c が導出できる。このとき, $M_{(1,3)} = \sin(b)$ から b が求まり, さらに, 変域に注意すれば, 残りの a と c も求まる。

2.9 分子の編集

編集には, 分子, 残基, 原子それぞれに対しての削除, コピー, 置換, 付加, 挿入, および結合の切断, 生成がある。このために dcore では, 主として (1) 共有結合の切断, (2) 共有結合の生成, (3) 1分子を2分子に分離, (4) 2分子を1分子に統合, (5) 分子のコピー, 等の機能を持っている。uif において, これら (1)~(5) の組み合わせと CAG の利用で編集を実現した。編集される部分構造は分子, 残基, 原子の如何にかかわらず内部ではすべて1分子とみなされ分子単位で操作される。なお置換, 付加, および挿入を行うために, アミノ酸や核酸塩基を部品分子として用意しており, それを使って新しい蛋白質や核酸を組み立てることができる。

編集においては原子間の結合関係を変えるだけで原子の座標は変化しないので, 長過ぎる共有結合になったり, 部分構造が幾何的に干渉することがある。そこで, 部分構造に対してリアルタイムで, 結合方向移動, 結合軸回りの回転, および結合角変化をさせて分子構造を修正する。このときも内部では, CAG を利用して, 部分構造の抽出を行う。

2.10 画面分割と俯瞰図

画面を最高 12 領域に分割し多画面表示をすることができる。このとき, 領域間で分子の複写や共有ができ, 3面図の生成と操作も可能である。

個々の領域に対して, より奥行きを出すためにデプスキューをかけたり遠近画法で表示したり, あるいは表示の拡大縮小をしたり, また前後クリップ面の位置を自由に設定することもできる。

さらに, 特別な領域として画面の奥行方向の情報と分子表示を大局的に把握する birdview ウィンドウを設けた。これは図6の右下部分に見られるように分子群の上部から見た概形, 視点の位置 (緑の EYE), 視角 (青線), 前後クリップ面 (赤線), xz 座標軸 (黄線) を表示し, 分子間の前後関係や表示範囲 (ズーム角) を確かめることができる。これらの設定を変えたりリアルタイムでそれに伴った表示に変更される。この機能により, 画面から分子が見えなくなってもどこにあるか分かり, 常時最適の位置での表示が可能になった。

2.11 バッチとログ機能

アニメーションを作成するためのバッチ機能を設けた。コマンドを記述したアニメ用バッチファイルを指定するとそれに従って, 分子の呼び込み, 編集, 表示変更, 移動, セーブ, および CAG の設定を実行する。また, バッチ処理の実行中に移動操作等を割り込んで行うこともできる。例えば, シミュレーション結果のバッチ処理中, これをいったん停止し別の角度から分子の動きや構造を観察できる。

バッチファイルで使われるコマンドは一種のインタプリタ言語で, 表1に示したような分子を操作するコマンド以外に, 条件分岐, 繰り返し, 変数への値の代入, および値の四則演算・論理演算・大小比較も用意されている。これらは yacc を用いて実現した。

また操作はすべてコマンドとして記録できる。コマンド入力はそのまま, またマウス入力是对応するコマンドに変換されログファイルに記録される。後にログファイルをバッチファイルとして指定すれば同じ操作手順を繰り返し実行できる。また, 既存のログファイルを編集すれば効率的にバッチが作成できる。

2.12 その他の機能

その他, (1) 分子データを分子構造データファイルから, もしくは直接 dcore から読み込み, 編集し, 書き込むためのユーザプログラム用 C 言語ライブラリ, (2) 画面を左右 2 領域に分割し, ステレオ表示¹⁵⁾



図 6 DNA の表示例

Fig. 6 DNA in space filling, transparent space filling, and ball-and-stick representation.



図 7 インシュリン様成長因子 I の表示例

Fig. 7 Insulin-like growth factor I in ribbon, cylinder, label, ball-and-stick, and transparent space filling representation.

を行う機能、(3)写真撮影のためのカラー輝度スケールと格子パターン生成、ハイライト部分の輝度調整の機能、(4)システムを終了したときの状態を環境ファイルに保存し次に起動したときと同じ状態に復帰する機能、(5)ユーザプログラムをメニューから実行できる機能がある。

3. 表示の実例

図6, 図7に表示例を示す。

図6は, DNA (PDB, 3BNA) である。らせんの一方を残基(ヌクレオチド)別の色分けで空間充填モデル表示を行い, 残りを半透明の空間充填モデルとボール&スティックモデルを重ね描きして表示した。片方のらせんの内側が観察でき, 他方のらせんとの相互的な結合が立体的に把握できる。

図7はインシュリン様成長因子I (PDB, 1GF1) である。これは, (1)残基別に色分けした線画, (2)半透明な空間充填モデルとボール&スティックの重ね描き, (3)原子別に色分けした空間充填モデル, (4)残基別に色分けした空間充填モデル, (5)残基別に色分けした半透明な空間充填モデル, (6)残基別に色分けしたボール&スティック, (7)ヘリックスの円筒, (8)ヘリックスを示す赤のリボン, (9)残基別に色分けした主鎖のリボン, (10)残基名, を同時に表示したものである。(2)の表示を行った残基は概形と化学構造が同時に分かる。(7)~(9)の表示は, 蛋白質の2次構造を明示するためである。このように, 任意の部位を任意のモデルで表示することにより構造の理解が容易になる。

4. おわりに

分子グラフィックスシステム LIVE の構造と特徴的な機能について述べた。論理演算によって選ばれたCAGを使い分子をリアルなイメージで様々な表現できた。また, ユーザインタフェースには, マウスによる2軸同時の回転・移動操作, 分子の位置や姿勢の数値表示, 俯瞰図の表示などの機能を持たせた。これにより高機能でユーザフレンドリーなシステムが構築できた。

将来の課題として, ユーザインタフェース uif は置き換えることが容易なことから, ポータビリティのよい X-Window 上にこれを移植したいと考えている。また, さらに有効な表示方法の開発, 他の計算機を利用する機能などを付け加え一層の充実を図る所存

である。

参考文献

- 1) 小出昭夫: 化学 CAD におけるコンピュータグラフィックス, 情報処理, Vol. 29, No. 10, pp. 1182-1189 (1988).
- 2) 佐藤昭彦: 分子設計支援ソフト, 日経コンピュータグラフィックス, No. 20, pp. 79-85, 日経マグローヒル, 東京 (1988).
- 3) 崎山文夫, 松原謙一(編): プロテイン エンジニアリング (化学増刊 113), 207 pp., 化学同人, 京都 (1988).
- 4) Fletterick, R. et al.: *Computer Graphics and Molecular Modeling, Current Communications in Molecular Biology*, 145 pp., Cold Spring Harbor Laboratory, New York (1986).
- 5) Ferrin, T. E. et al.: The MIDAS Database System, *J. Mol. Graph.*, Vol. 6, pp. 2-12 (1988).
- 6) Ferrin, T. E. et al.: The MIDAS Display System, *J. Mol. Graph.*, Vol. 6, pp. 13-27 (1988).
- 7) Afshar, M. et al.: PCDRA: PC Interactive Molecular Representation and Modeling System, *J. Mol. Graph.*, Vol. 8, pp. 39-44 (1990).
- 8) Vriend, G.: WHAT IF: A Molecular Modeling and Drug Design Program, *J. Mol. Graph.*, Vol. 8, pp. 52-56 (1990).
- 9) Burkert, U. and Allinger, N. L. (大澤映二, 竹内敬人(訳)): 分子力学, 342 pp., 啓学出版, 東京 (1986).
- 10) 田中 實, 山本良一(編): 計算物理学と計算化学, 321 pp., 海文堂, 東京 (1988).
- 11) 岡田 勲, 大澤映二(編): 分子シミュレーション入門, 227 pp., 海文堂, 東京 (1989).
- 12) Bernstein, F. C. et al.: The Protein Data Bank. A Computer-based Archival File for Macromolecular Structures, *J. Mol. Biol.*, Vol. 112, pp. 535-542 (1977).
- 13) Kernighan, B. W. and Pike, R. (石田晴久(監訳)): UNIX プログラミング環境, 514 pp., アスキー, 東京 (1985).
- 14) Harrington, S. (郡山 彬(訳)): コンピュータグラフィックス [I, II], 592 pp., マグロウヒル, 東京 (1988).
- 15) 加納 裕, 斎藤史彦: 立体映像とコンピュータグラフィックス, 情報処理, Vol. 31, No. 2, pp. 265-273 (1990).

(平成2年8月20日受付)

(平成2年12月18日採録)

**石井 隆寛**

1964年生。1988年東京大学工学部航空学科卒業。1990年同学大学院農学系研究科応用生命工学専攻修士課程修了。現在東京大学農学部助手。生体分子のシミュレーションに関する研究に従事。

**池口 満徳**

1967年生。1989年東京大学工学部計数工学科卒業。1991年同学大学院農学系研究科応用生命工学専攻修士課程修了。現在同専攻博士課程在学中。タンパク質の構造解析に関する研究に従事。

**山田 俊宏**

1966年生。1989年東京大学工学部反応化学科卒業。1991年同学大学院農学系研究科応用生命工学専攻修士課程修了。タンパク質の構造予測に関する研究に従事。現在東レ(株)勤務。

**土井 淳多 (正会員)**

1959年東京大学農学部卒業。同学大学院博士課程を経て1966年同学農学部助教授。1978年同教授。現在、同学部応用生命工学専攻生物情報工学講座担任。日本機械学会、可視化情報学会、Molecular Graphics Society 等各会員。