

# 使用 MolAICal 生成肽的教程

Qifeng Bai

邮箱: [molaical@yeah.net](mailto:molaical@yeah.net)

主页: <https://molaical.github.io> 或 <https://molaical.gitlab.io>

## 1. 介绍

本部分总结了三种常用于表示肽和生物聚合物信息的格式: BILN (生物聚合物线性符号) [1]、HELM (大分子分层编辑语言) [2] 和 FASTA。每种格式有不同的设计目标和典型用途: 一种是轻量级且人类可读的, 一种是大分子的正式标准, 另一种是生物信息学中广泛使用的最小序列容器。下面描述了每种格式的定义特征、典型用例、优势和局限性, 并给出一个清晰的示例:

**FASTA** 是表示核苷酸序列或氨基酸 (蛋白质) 序列的传统基于文本的格式。它是生物信息学工具 (如 BLAST) 和简单序列存储的通用标准。**局限性:** FASTA 本质上无法描述:

- ◆ 环化 (拓扑结构)。
- ◆ 分支或翻译后修饰 (除非使用非标准的变通方法)。
- ◆ 复杂的化学连接物或非天然氨基酸 (除非模糊地使用'X')。

BILN 和 HELM 可以解决 FASTA 的局限性。BILN 是 MolAICal 中的默认格式。本教程将展示使用 MolAICal 生成肽的示例。

## 2. 材料

### 2.1. 软件需求

1. **MolAICal:** <https://molaical.github.io>
2. **PyMol:** <https://github.com/cgohlke/pymol-open-source-wheels>,  
<https://github.com/maabuu/pymol-wheels>, <https://pypi.org/project/pymol-open-source-whl/>, <https://github.com/cnpeim/PyMOL4Win>

### 2.2. 示例文件

1. 所有必要的教程文件可从以下地址下载:  
<https://gitee.com/molaical/tutorials/tree/master/027-peptideGen>

## 3. 操作步骤

### 3.1. 生成线性肽

打开教程材料文件夹 "027-peptideGen":

```
#> cd 027-peptideGen
```

首先，用户可以通过以下命令查看 MolAICal 中肽生成的更多帮助信息：

```
#> molaical.exe -call run -c pepgen -i --help
```

1) 生成 10 个具有 8 个残基、带有 C 端和 N 端封端但不含 3D 结构的随机线性肽。

```
#> molaical.exe -call run -c pepgen -i -l 8 -n 10 -o sequences.txt
```

2) 如果除了肽序列（不含 C 端和 N 端封端）外还需要 3D 结构，请运行以下命令：

```
#> molaical.exe -call run -c pepgen -i -l 8 -n 10 -3d -nc -o sequences.txt -dir results
```

注意：

- ◆ **-c:** 如果其值为"pepgen"，则会生成肽
- ◆ **-i:** 程序的命令参数。它包含外部程序的所有命令行参数，但'-i'参数必须是最后指定的，而 MolAICal 的所有其他标识符（例如'-p'等）必须放在'-i'之前。
- ◆ **-l:** 肽中氨基酸残基的数量
- ◆ **-n:** 要生成的唯一序列数量
- ◆ **-o:** 生成序列的输出文件；默认文件名为"peptide\_sequences.txt"。
- ◆ **-3d:** 使用 pyPept 生成 3D 结构
- ◆ **-dir:** 3D 结构的输出目录；默认文件夹名为"output\_3d"。
- ◆ **-nc:** 不向序列添加乙酰基/酰胺封端

对于每个肽，MolAICal 生成 5 种类型的肽，包括基础(base)、 $\alpha$ -螺旋(alpha-helix)、 $\beta$ -折叠(beta-sheet)、 $\beta$ -桥(beta-bridge)、转角-卷曲(turn-coil)和 RDKit 格式。例如，"IVASETEE\_turn-coil.pdb"，前缀"IVASETEE"是该肽的序列。"turn-coil"是结构类型。

### 3.2. 生成环状肽

生成环状肽；然而，环状肽仅在"base"和"turn-coil"格式中受支持，因此参数"-ss alpha-helix beta-sheet"将被忽略，运行如下命令：

```
#> molaical.exe -call run -c pepgen -i -l 8 -n 3 -cc -3d -c 4 -rc 3 -nc
```

注意：

- ◆ **-cc:** 生成环状肽（仅 BILN/HELM）
- ◆ **-c:** 用于并行处理的 CPU 核心数（此"-c"在"-i"之后）。
- ◆ **-nc:** 不向序列添加乙酰基/酰胺封端
- ◆ **-rc:** 要生成的 RDKit 随机构象数量（默认值：1）

通过 PyMol 从目录"output\_3d"中打开一个环状肽（见图 1）。PyMol 不会显示环状卡通肽。用户可以显示线模式来查看环状肽：点击"S"然后选择"lines"（见图 1）。



图 1

### 3.3. 基于固定长度修改肽

随机修改长度为 16 的肽段，使用其他氨基酸残基，同时保持位置 1 和 3 的氨基酸残基不变，运行如下命令：

```
#> molaical.exe -call run -c pepgen -i -l 16 -n 10 -s CCCCCCCCCCC -d none -fix 1 3 -f biln -3d -nc -c 10
```

注意：

- ◆ **-s:** 起始序列（可选）
- ◆ **-d:** 从种子序列扩展的方向。如果"-d"的值为'none'，MolAICal 将不会扩展肽。
- ◆ **-fix:** 生成过程中保持不变的位置（1 为起始索引）
- ◆ **-f:** 序列的输出格式（默认：biln）

关于肽的更多生成方式，请参阅 MolAICal 手册。

# 参考文献

1. Jiménez-García, B.; Roel-Touris, J.; Romero-Durana, M.; Vidal, M.; Jiménez-González, D.; Fernández-Recio, J., LightDock: a new multi-scale approach to protein-protein docking. *Bioinformatics* 2018, 34 (1), 49-55.
2. Krishnanand, K. N.; Ghose, D., Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions. *Swarm Intelligence* 2009, 3 (2), 87-124.