**Moltemplate Guide 1**

<https://www.youtube.com/watch?v=rBHyWjWVyaA>

**Moltemplate installation**

1. 리눅스 Bash쉘을 킨다
2. git clone <https://github.com/jewettaij/moltemplate>
3. cd moltemplate
4. pip3 install .
5. export PATH=”$PATH:<다운로드받은디렉터리>/moltemplate/moltemplate”
6. export PATH=”$PATH:<다운로드받은디렉터리>/moltemplate/moltemplate/scripts”

**Moltemplate example**

1. cd moltemplate
2. cd all\_atom/force\_field\_OPLSAA/
3. cd butane

**step 1) bash README\_setup.sh**

vi README\_setup.sh & and see..

1. cd moltemplate\_files
2. moltemplate.sh system.lt
3. mv -f system.data system.in\* ../

cd moltemplate\_files  
vi system.lt & and see…

import “butane.lt”

vi butane.lt & and see…

import “ch2group.lt”  
 import “ch3group.lt”

bash README\_setup.sh & and wait…

**step 2) bash README\_run.sh**

example lammps run execute

**Moltemplate Guide 2**

**mol22lt.py 파일을 사용해서 .lt형식으로 분자 구조를 바꾸는 방법**

**Moltemplate Guide 3**

<https://github.com/jewettaij/moltemplate/blob/master/moltemplate/force_fields/oplsaa.lt>

<https://github.com/jewettaij/moltemplate/blob/master/moltemplate/force_fields/oplsaa2023_original_format/Jorgensen_et_al-2023-The_Journal_of_Physical_Chemistry_B.sup-2.par>

**oplsaa.lt 보는 방법**

OPLSAA.lt파일 내부를 보면 아래와 같다.

텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

여기서 특정 기능기 환경에서 어떤 원소에 대한 식별자 번호를 저장하고 있는 것을 볼 수 있다. 이때 내가 모델링하려는 분자의 원소에 해당하는 기능기 환경 설명을 찾고 그 번호를 기억해 나의 분자의 .lt파일에 기재해주면 된다. 텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

기본적으로 mol22lt.py를 실행하면 위와 같이 파일이 생성된다. 이때 C.ar은 OPLSAA의 설정이 아니므로 이를 OPLSAA상의 번호로 바꿔준다.

그리고 최상단에 import “oplsaa.lt”를 추가한다

또한 위 블록을 감싸는 부분에

UNL1 inherits OPLSAA {

를 추가한다.

원문 <https://docs.lammps.org/Howto_moltemplate.html>

**8.6.4. Moltemplate 튜토리얼**

이 튜토리얼에서는 Moltemplate 도구를 사용하여 OPLS-AA 힘장(Force Field)을 이용한 고전적 분자동역학(MD) 시뮬레이션을 설정하는 방법을 다룹니다.

첫 번째 작업은 유기 화합물을 정의하고 LAMMPS용 완전한 입력 파일을 생성하는 것입니다.

두 번째 작업은 외부 도구(예: PACKMOL)로 생성된 분자 샘플을 PDB 파일로 내보내고, OPLS-AA 힘장을 매핑하는 것입니다.

이 튜토리얼에서 사용되는 파일들은 LAMMPS 소스 코드 배포판의 tools/moltemplate/tutorial-files 폴더에서 찾을 수 있습니다.

**유기 용매 시뮬레이션**

이 예제에서는 유기 용매 포름아미드(Formamide, HCONH₂)의 정육면체 박스를 생성하는 방법을 설명합니다.

**1. LAMMPS-템플릿(LT) 파일 형식으로 분자 구조 정의**

첫 번째 단계는 LT(LAMMPS-template) 파일 형식으로 분자 구조를 정의하는 것입니다.

이 파일은 Moltemplate 객체 \_FAM에 저장되며, 이는 다음과 같이 선언됩니다.



**설명:**

* \_FAM 객체는 기존의 OPLSAA 객체를 상속받습니다.
* OPLSAA 객체는 OPLS-AA 힘장 정보(힘장 매개변수, 원자 유형 정의, 부분 전하, 질량, 결합-각도 규칙 등)를 포함하고 있습니다.

**2. Data Atoms 섹션 작성**

원자 구조(Atomic Structure)는 LAMMPS 데이터 파일의 Atoms 섹션을 채우는 역할을 하는 write('Data Atoms') {} 명령어에 입력됩니다.

텍스트, 폰트, 스크린샷, 대수학이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**설명:**

LAMMPS에서 OPLS-AA 힘장은 atom\_style full을 사용하므로 다음과 같은 형식을 따릅니다



원자 ID는 Moltemplate $-type 변수를 사용하여 자동으로 생성됩니다.

molID도 같은 변수를 사용하여 분자 전체에 동일한 값이 할당됩니다.

원자 유형은 @-type 변수를 사용하여 할당됩니다.

예를 들어,

@atom:177 → OPLS-AA 힘장에서 해당하는 탄소 원자 유형

@atom:178 → OPLS-AA 힘장에서 해당하는 산소 원자 유형

**3. Data Bond List 섹션 작성**

각 원자 간 결합(Bond) 정보를 정의해야 합니다.

텍스트, 스크린샷, 폰트, 대수학이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**설명:**

LAMMPS에서 결합 정보는 Bond List에 정의됩니다.

각 결합(Bond)은 $bond:C1, $bond:C2 등의 변수로 정의되며, 이는 Moltemplate가 자동으로 숫자로 변환합니다.

예를 들어,

$bond:C1 → C00 (탄소) - O01 (산소) 간 결합

$bond:C3 → C00 (탄소) - N02 (질소) 간 결합

**4. 전하(Charge)와 결합(Bond) 타입 자동 생성**

이 예제에서는 전하(charge)를 직접 지정할 필요가 없습니다.

OPLS-AA 힘장은 각 원자 유형(atom type)에 따라 자동으로 전하를 할당합니다.

마찬가지로, 각 원자 유형이 결합(bond) 유형을 결정하므로 별도의 Bond Coefficients 정의가 필요하지 않습니다.

즉, 각도(Angle), 디하이드랄(Dihedral), 비틀림(Impropers) 등의 다른 결합 상호작용도 Moltemplate에 의해 자동으로 생성됩니다.

**최종 formamide.lt 파일 구조**

텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**결론**

Moltemplate를 사용하여 LAMMPS에서 OPLS-AA 힘장을 적용한 유기 용매(포름아미드) 시뮬레이션을 설정할 수 있습니다.

원자 유형과 결합 유형은 Moltemplate가 자동으로 OPLS-AA 힘장을 기반으로 생성하므로 사용자가 따로 정의할 필요가 없습니다.

결합, 각도, 디하이드랄 등은 원자 유형에 따라 자동 생성됩니다.