分子动力学文件语法高亮 VScode 插件 v1.0

源程序文档

2025 年 9 月 9 日

一、 扩展配置模块

1.1 package.json - 扩展主配置文件

```json
{
  "name": "md-highlighter",
  "displayName": "md-highlighter",
  "description": "Highlighting files about molecular modeling and dynamics simulation.",
  "version": "0.0.7",
  "publisher": "gxf1212",
  "repository": "https://github.com/gxf1212/md-highlighter",
  "engines": {
    "vscode": "^1.70.0"
  },
  "icon": "./images/benzene.png",
  "categories": [
    "Programming Languages"
  ],
  "contributes": {
    "languages": [
      {
        "id": "rtf",
        "aliases": [
          "NAMD force field",
          "rtf"
        ],
        "extensions": [
          ".rtf",
          ".str",
          ".inp",
          ".prm",
          ".par"
        ],
        "configuration": "./language-configuration.json"
      },
      {
        "id": "pdb",
        "aliases": [
          "PDB file",
          "pdb"
        ],
        "extensions": [
          ".pdb"
        ],
        "configuration": "./language-configuration.json"
      },
      {
        "id": "psf",
        "aliases": [
          "NAMD topology file"
        ],
        "extensions": [
          ".psf"
        ],
        "configuration": "./language-configuration.json"
```

```json
  },
  {
    "id": "in",
    "aliases": [
      "Amber in file"
    ],
    "extensions": [
      ".in"
    ],
    "filenamePatterns": [
      "leaprc.*"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "prmtop",
    "aliases": [
      "Amber topology file",
      "prmtop"
    ],
    "extensions": [
      ".prmtop",
      ".parm7"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "inpcrd",
    "aliases": [
      "Amber coordinate file",
      "inpcrd"
    ],
    "extensions": [
      ".inpcrd",
      ".rst7"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "prepin",
    "aliases": [
      "Amber prepin file",
      "prepin"
    ],
    "extensions": [
      ".prepin",
      ".prepi"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "frcmod",
    "aliases": [
      "Amber parameter file",
      "frcmod"
    ],
    "extensions": [
```

```json
          ".frcmod"
        ],
        "filenamePatterns": [
          "frcmod.*",
          "parm*.dat",
          "*parm.dat",
          "gaff*.dat",
          "lipid*.dat",
          "GLYCAM*.dat"
        ],
        "configuration": "./language-configuration.json"
      },
      {
        "id": "lib",
        "aliases": [
          "Amber topology type file"
        ],
        "extensions": [
          ".lib",
          ".off"
        ],
        "configuration": "./language-configuration.json"
      },
      {
        "id": "ac",
        "aliases": [
          "Antechamber file"
        ],
        "extensions": [
          ".ac",
          ".AC"
        ],
        "configuration": "./language-configuration.json"
      },
      {
        "id": "mc",
        "aliases": [
          "Antechamber mainchain file"
        ],
        "extensions": [
          ".mc"
        ],
        "configuration": "./language-configuration.json"
      },
      {
        "id": "gro",
        "aliases": [
          "Gromacs coordinate file"
        ],
        "extensions": [
          ".gro"
        ],
        "configuration": "./language-configuration.json"
      },
      {
        "id": "atp",
        "aliases": [
```

```
      "Gromacs atomtype file"
    ],
    "extensions": [
      ".atp"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "arn",
    "aliases": [
      "Gromacs atom renaming specification"
    ],
    "extensions": [
      ".arn"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "rtp",
    "aliases": [
      "Gromacs residue type file"
    ],
    "extensions": [
      ".rtp"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "hdb",
    "aliases": [
      "Gromacs hydrogen database"
    ],
    "extensions": [
      ".hdb"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "r2b",
    "aliases": [
      "Gromacs r2b file"
    ],
    "extensions": [
      ".r2b"
    ],
    "configuration": "./language-configuration.json"
  },
  {
    "id": "tdb",
    "aliases": [
      "Gromacs terminal database"
    ],
    "extensions": [
      ".tdb"
    ],
    "configuration": "./language-configuration.json"
  },
```

```json
    {
      "id": "mtp",
      "aliases": [
        "Gromacs mutation topology file"
      ],
      "extensions": [
        ".mtp"
      ],
      "configuration": "./language-configuration.json"
    },
    {
      "id": "vsd",
      "aliases": [
        "Gromacs virtual site definition file"
      ],
      "extensions": [
        ".vsd"
      ],
      "configuration": "./language-configuration.json"
    },
    {
      "id": "mol2",
      "aliases": [
        "mol2 file"
      ],
      "extensions": [
        ".mol2"
      ],
      "configuration": "./language-configuration.json"
    },
    {
      "id": "sdf",
      "aliases": [
        "sdf file"
      ],
      "extensions": [
        ".sdf",
        ".mol"
      ],
      "configuration": "./language-configuration.json"
    },
    {
      "id": "plumed",
      "aliases": [
        "plumed input file",
        "plumed"
      ],
      "extensions": [
        ".plumed.dat"
      ],
      "configuration": "./language-configuration.json"
    }
  ],
  "grammars": [
    {
      "language": "rtf",
      "scopeName": "source.rtf",
```

```
      "path": "./syntaxes/rtf.tmLanguage.json"
    },
    {
      "language": "pdb",
      "scopeName": "source.pdb",
      "path": "./syntaxes/pdb.tmLanguage.json"
    },
    {
      "language": "psf",
      "scopeName": "source.psf",
      "path": "./syntaxes/psf.tmLanguage.json"
    },
    {
      "language": "in",
      "scopeName": "source.in",
      "path": "./syntaxes/in.tmLanguage.json"
    },
    {
      "language": "prmtop",
      "scopeName": "source.prmtop",
      "path": "./syntaxes/prmtop.tmLanguage.json"
    },
    {
      "language": "inpcrd",
      "scopeName": "source.inpcrd",
      "path": "./syntaxes/inpcrd.tmLanguage.json"
    },
    {
      "language": "prepin",
      "scopeName": "source.prepin",
      "path": "./syntaxes/prepin.tmLanguage.json"
    },
    {
      "language": "frcmod",
      "scopeName": "source.frcmod",
      "path": "./syntaxes/frcmod.tmLanguage.json"
    },
    {
      "language": "lib",
      "scopeName": "source.lib",
      "path": "./syntaxes/lib.tmLanguage.json"
    },
    {
      "language": "ac",
      "scopeName": "source.ac",
      "path": "./syntaxes/ac.tmLanguage.json"
    },
    {
      "language": "mc",
      "scopeName": "source.mc",
      "path": "./syntaxes/mc.tmLanguage.json"
    },
    {
      "language": "gro",
      "scopeName": "source.gro",
      "path": "./syntaxes/gro.tmLanguage.json"
    },
```

```json
    {
      "language": "atp",
      "scopeName": "source.atp",
      "path": "./syntaxes/atp.tmLanguage.json"
    },
    {
      "language": "arn",
      "scopeName": "source.arn",
      "path": "./syntaxes/arn.tmLanguage.json"
    },
    {
      "language": "rtp",
      "scopeName": "source.rtp",
      "path": "./syntaxes/rtp.tmLanguage.json"
    },
    {
      "language": "hdb",
      "scopeName": "source.hdb",
      "path": "./syntaxes/hdb.tmLanguage.json"
    },
    {
      "language": "r2b",
      "scopeName": "source.r2b",
      "path": "./syntaxes/r2b.tmLanguage.json"
    },
    {
      "language": "tdb",
      "scopeName": "source.tdb",
      "path": "./syntaxes/tdb.tmLanguage.json"
    },
    {
      "language": "mtp",
      "scopeName": "source.mtp",
      "path": "./syntaxes/mtp.tmLanguage.json"
    },
    {
      "language": "vsd",
      "scopeName": "source.vsd",
      "path": "./syntaxes/vsd.tmLanguage.json"
    },
    {
      "language": "mol2",
      "scopeName": "source.mol2",
      "path": "./syntaxes/mol2.tmLanguage.json"
    },
    {
      "language": "sdf",
      "scopeName": "source.sdf",
      "path": "./syntaxes/sdf.tmLanguage.json"
    },
    {
      "language": "plumed",
      "scopeName": "source.plumed",
      "path": "./syntaxes/plumed.tmLanguage.json"
    }
  ]
},
```

```json
  "__metadata": {
    "id": "ea9482d2-d0dc-42fc-bbb3-a5e5f99d95ab",
    "publisherId": "bacd9c21-e15d-4c96-a8e9-e7b218fadd67",
    "publisherDisplayName": "XufanGao",
    "targetPlatform": "undefined",
    "updated": false,
    "isPreReleaseVersion": false,
    "hasPreReleaseVersion": false,
    "installedTimestamp": 1706966519959,
    "pinned": false
  }
}
```

1.2 language-configuration.json - 语言通用配置

```json
{
    "comments": {
        // symbol used for single line comment. Remove this entry if your language does
    not support line comments
        "lineComment": "!",
        // symbols used for start and end a block comment. Remove this entry if your
    language does not support block comments
        "blockComment": [ "/*", "*/" ]
    },
    // symbols used as brackets
    "brackets": [
        ["{", "}"],
        ["[", "]"],
        ["(", ")"]
    ],
    // symbols that are auto closed when typing
    "autoClosingPairs": [
        ["{", "}"],
        ["[", "]"],
        ["(", ")"],
        ["\"", "\""],
        ["'", "'"]
    ],
    // symbols that can be used to surround a selection
    "surroundingPairs": [
        ["{", "}"],
        ["[", "]"],
        ["(", ")"],
        ["\"", "\""],
        ["'", "'"]
    ]
}
```

二、 NAMD/CHARMM 格式支持模块

2.1 PDB 文件语法定义

```json
{
  "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage.
    json",
  "name": "PDB file syntax highlighting",
  "scopeName": "source.pdb",
  "fileTypes": [
    "pdb"
  ],
  "patterns": [
    {
      "begin": "^(CRYST|REMARK|HEADER|TITLE|COMPND|SOURCE|KEYWDS|EXPDTA|AUTHOR|REVDAT|
    JRNL|ORIGX|SCALE\\S*|DBREF|CONECT|MASTER)\\b",
      "end": "$",
      "name": "comment.line.remark.pdb"
    },
    {
      "//": "not for MODEL ENDMDL",
      "name": "keyword.other.record-name.pdb",
      "match": "\\b(CRYST1|TER|END|HELIX|SHEET|SSBOND)\\b"
    },
    {
      "begin": "^ATOM  |^HETATM",
      "end": "$",
      "name": "keyword.atom.pdb",
      "patterns": [
        {
          "name": "constant.numeric.atom-number.pdb",
          "match": "(?<=^.{6}).{5}"
        },
        {
          "name": "entity.name.tag.atom-name.pdb",
          "match": "(?<=^.{12}).{4}"
        },
        {
          "name": "string.unquoted.alternate-location.pdb",
          "match": "(?<=^.{16}).{1}"
        },
        {
          "name": "entity.name.type.residue-name.pdb",
          "match": "(?<=^.{17}).{4}"
        },
        {
          "name": "support.type.chain-id.pdb",
          "match": "(?<=^.{21}).{1}"
        },
        {
          "name": "variable.other.residue-number.pdb",
          "match": "(?<=^.{22}).{4}"
        },
        {
          "name": "entity.name.function.code-for-insertions-of-residues.pdb",
          "match": "(?<=^.{26}).{1}"
        },
        {
          "name": "constant.numeric.x-coordinate.pdb",
          "match": "(?<=^.{30}).{8}"
```

```
      },
      {
        "name": "constant.numeric.y-coordinate.pdb",
        "match": "(?<=^.{38}).{8}"
      },
      {
        "name": "constant.numeric.z-coordinate.pdb",
        "match": "(?<=^.{46}).{8}"
      },
      {
        "name": "string.quoted.occupancy.pdb",
        "match": "(?<=^.{54}).{6}"
      },
      {
        "name": "storage.type.temperature-factor.pdb",
        "match": "(?<=^.{60}).{6}"
      },
      {
        "name": "support.type.segment-name.pdb",
        "match": "(?<=^.{72}).{4}"
      },
      {
        "name": "string.quoted.element-symbol.pdb",
        "match": "(?<=^.{76}).{2}"
      },
      {
        "name": "variable.other.charge.pdb",
        "match": "(?<=^.{78}).{2}"
      }
    ]
  }
 ]
}
```

2.2 RTF 文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "NAMD force field (atom, residue, parameter) file",
    "scopeName": "source.rtf",
    "fileTypes": [
        "rtf",
        "inp",
        "str",
        "prm"
    ],
    "patterns": [
        {
            "//": "starting with exclamation",
            "name": "comment.line.exclamation",
            "match": "!.*$"
        },
        {
            "name": "comment.line.asterisk",
```

```
            "match": "^\\s*\\*.*"
        },
        {
            "name": "keyword.rtf",
            "match": "\\b(MASS|RESI|PRES|DELETE|ATOM|BOND|ANGLE(S)?|DIHE(DRAL)?|IMPR|IMPH
|(?i)IC|BILD|ACCE(PTOR)?|DONO(R)?|DOUBLE|DOUB|(?i)PATC(H)?|PATCHING|LONEPAIR|COLI|
DIST|SCAL|DECL|DEFA|AUTO|FIRS|LAST|first|last|default)\\b"
        },
        {
            "name": "keyword.prm",
            "match": "(MASS|ATOMS|BONDS|ANGLES|DIHEDRALS|IMPROPER(S)?|CMAP|NONBONDED|
NBFIX|HBOND|CUTHB)"
        },
        {
            "name": "entity.name.section.prm",
            "match": "\\b(BONDS|ANGLES|DIHEDRALS|IMPROPERS|CMAP|NONBONDED|NBFIX)\\b"
        },
        {
            "name": "support.type.end.rtf",
            "match": "^END|^GROUP"
        },
        {
            "//": "floats in lines starting with those keywords",
            "name": "constant.numeric.rtf",
            "match": "(?<=\\b(MASS|ATOM|RESI|PRES|IC|BILD|LONEPAIR)\\b.*?)\\b(-?\\d+\\.
\\d*)\\b"
        },
        {
            "//": "the number after MASS, all -1, or 1,2,3...",
            "name": "string.mass_number.rtf",
            "match": "(?<=\\bMASS\\b.*?)\\s+-?\\d+\\s+"
        },
        {
            "//": "element symbol in MASS line",
            "name": "string.quoted.element-symbol.rtf",
            "match": "(?<=^MASS\\s+-?\\d+\\s+\\S+\\s+\\b(-?\\d+\\.\\d*)\\b\\s+)\\S+"
        },
        {
            "//": "the word after RESI|PRES",
            "name": "entity.name.type.resname.rtf",
            "match": "(?<=\\b(RESI|PRES)\\s)\\S+"
        },
        {
            "//": "the number after RESI|PRES resname",
            "name": "string.quoted.charge.rtf",
            "match": "(?<=\\b(RESI|PRES)\\s+\\S+\\s+)-?\\d+.\\d+"
        },
        {
            "//": "the several words after a keyword",
            "note": "bond: variable length of pairs; impr: four strings",
            "////": "Maybe atom names can be de",
            "name": "entity.name.tag.atom.rtf",
            "match": "(?<=\\bATOM\\s*)\\S+|(?<=\\bBOND\\s*)(\\S+\\s+\\S+\\s*)+|(?<=
\\bANGLE\\s*)((\\S+\\s*){3})+|(?<=\\bIMPR\\s*)((\\S+\\s*){4})+|(?<=\\bIMPH\\s*)((\\S+
\\s*){4})+|(?<=\\bCMAP\\s*)((\\S+\\s*){8})+|(?<=\\b\\s*(?i)IC\\s*)(\\S+\\s+){4}|(?<=
\\b\\s*BILD\\s*)(\\S+\\s+){4}"
        },
```

```
        {
            "//": "the several words after a keyword.",
            "name": "entity.name.tag.atom.rtf",
            "match": "(?<=\\bACCE(PTOR)?\\s*)(\\S+\\s*)+|(?<=\\bDONO(R)?\\s*)(\\S+\\s*)
+|(?<=\\bDOUBLE\\s*)(\\S+\\s+\\S+\\s*)+|(?<=\\bPATC\\s*)(\\S+\\s+){4}|(?<=\\bPATCH\\s
*)(\\S+\\s+){4}"
        },
        {
            "//": "rarer keywords",
            "name": "entity.name.tag.atom.rtf",
            "match": "(?<=\\bDECL\\s*)(?:[+-]\\S+\\s*)|(?<=\\bDEFA\\s*)((\\S+\\s*){4})
+|(?<=\\bLONEPAIR\\s+\\S+\\s+)\\S+\\s+\\S+\\s+\\S+"
        },
        {
            "//": "all floats, charge, mass, etc. put this before atom type. integers
like dihedral n is not here",
            "//": "with \\b( )\\b, cannot match minus sign...",
            "//": "after charge, before atom-type.prm, otherwise overwritten...",
            "name": "constant.numberic.rtf",
            "match": "-?\\d+\\.\\d*"
        },
        {
            "//": "dihedral n",
            "name": "string.quoted.dihedral-n.prm",
            "match": "(?<=-?\\d+\\.\\d*\\s+)\\d(?=\\s+-?\\d+\\.\\d*)"
        },
        {
            "//": "all strings made up of capitalized letters (in special files there are
 lower case letters) and numbers. matching a lot of remaining strings...put them
before this!",
            "name": "support.type.atom-type.prm",
            "match": "\\b([A-Za-z0-9]+)\\b"
        },
        {
            "//": "a special type",
            "name": "support.type.atomtype.prm",
            "match": "\\b(none)\\b|\\b(NONE)\\b"
        }
    ]
}
```

2.3 PSF 文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "PSF from GUI, that from vmd is different",
    "scopeName": "source.psf",
    "fileTypes": [
        "psf"
    ],
    "patterns": [
        {
            "name": "comment.line.psf",
            "match": "^\\\*.*$"
```

```
        },
        {
            "begin": "^\\s+REMARKS\\b",
            "end": "$",
            "name": "comment.line.remark.pdb"
        },
        {
            "//": "ATOM and BOND is missing...",
            "name": "keyword.psf",
            "match": "\\b(PSF|CMAP|NTITLE|NATOM|NBOND|NTHETA|NPHI|NIMPHI|NDON|NACC|NNB|
NGRP|NUMLP|NCRTERM)\\b"
        },
        {
            "name": "entity.name.function.total_number",
            "match": "(?<=^\\s+)\\d+(?= !N)"
        },
        {
            "begin": "(!NATOM\\s*$)",
            "end": "(!NBOND)",
            "name": "keyword.atom.psf",
            "//": "atom names use \\S+ instead of \\b[A-Z0-9]+\\b to include names with
'",
            "//": "format: atomid segname resid resname atomname atomtype charge mass
unknown",
            "//": "^\\s+\\d+\\s+\\S+\\s+\\d+\\s+\\S+\\s+\\S+\\s+\\b[A-Z0-9]+\\b\\s+(-?\\d
+\\.\\d*(E-01)?)\\s+(-?\\d+\\.\\d*)\\s+\\d+$",
            "patterns": [
                {
                    "//": "NATOM still not solved. see my notes",
                    "name": "keyword.other.psf",
                    "match": "NATOM|NBOND"
                },
                {
                    "name": "entity.name.function.total_number",
                    "match": "(?<=^\\s+)\\d+(?= !N)"
                },
                {
                    "name": "constant.numeric.atom-number.psf",
                    "match": "(?<=^\\s+)\\d+(?=\\s+\\S+\\s+\\d+\\s+\\S+\\s+\\S+\\s+\\b[A-
Z0-9]+\\b\\s+(-?\\d+\\.\\d*(E-01)?)\\s+(-?\\d+\\.\\d*)\\s+\\d+$)"
                },
                {
                    "name": "variable.other.segment-name.psf",
                    "match": "(?<=^\\s+\\d+\\s+)\\S+(?=\\s+\\d+\\s+\\S+\\s+\\S+\\s+\\b[A-
Z0-9]+\\b\\s+(-?\\d+\\.\\d*(E-01)?)\\s+(-?\\d+\\.\\d*)\\s+\\d+$)"
                },
                {
                    "name": "support.type.residue-number.psf",
                    "match": "(?<=^\\s+\\d+\\s+\\S+\\s+)\\d+(?=\\s+\\S+\\s+\\S+\\s+\\b[A-
Z0-9]+\\b\\s+(-?\\d+\\.\\d*(E-01)?)\\s+(-?\\d+\\.\\d*)\\s+\\d+$)"
                },
                {
                    "name": "entity.name.type.residue-name.psf",
                    "match": "(?<=^\\s+\\d+\\s+\\S+\\s+\\d+\\s+)\\S+(?=\\s+\\S+\\s+\\b[A-
Z0-9]+\\b\\s+(-?\\d+\\.\\d*(E-01)?)\\s+(-?\\d+\\.\\d*)\\s+\\d+$)"
                },
                {
```

```
                "name": "entity.name.tag.atom-name.psf",
                "match": "(?<=^\\s+\\d+\\s+\\S+\\s+\\d+\\s+\\S+\\s+)\\S+(?=\\s+\\b[A-
Z0-9]+\\b\\s+(-?\\d+\\.\\d*(E-01)?)\\s+(-?\\d+\\.\\d*)\\s+\\d+$)"
            },
            {
                "name": "support.type.atom-type.psf",
                "match": "(?<=^\\s+\\d+\\s+\\S+\\s+\\d+\\s+\\S+\\s+\\S+\\s+)\\b[A-Z
0-9]+\\b(?=\\s+(-?\\d+\\.\\d*(E-01)?)\\s+(-?\\d+\\.\\d*)\\s+\\d+$)"
            },
            {
                "name": "string.quoted.charge.psf",
                "match": "(?<=^\\s+\\d+\\s+\\S+\\s+\\d+\\s+\\S+\\s+\\S+\\s+\\b[A-Z
0-9]+\\b\\s+)(-?\\d+\\.\\d*(E-01)?)(?=\\s+(-?\\d+\\.\\d*)\\s+\\d+$)"
            },
            {
                "name": "constant.numeric.atomic-mass.psf",
                "match": "(?<=^\\s+\\d+\\s+\\S+\\s+\\d+\\s+\\S+\\s+\\S+\\s+\\b[A-Z
0-9]+\\b\\s+(-?\\d+\\.\\d*(E-01)?)\\s+)(-?\\d+\\.\\d*)(?=\\s+\\d+$)"
            },
            {
                "name": "comment.line.unknown.psf",
                "match": "(?<=^\\s+\\d+\\s+\\S+\\s+\\d+\\s+\\S+\\s+\\S+\\s+\\b[A-Z
0-9]+\\b\\s+(-?\\d+\\.\\d*(E-01)?)\\s+(-?\\d+\\.\\d*)\\s+)\\d+(?=$)"
            }
        ]
    },
    {
        "name": "constant.numeric.psf",
        "match": "\\b\\d+\\b|(?<=^\\s*)\\d+"
    }
    ]
}
```

三、 Amber 格式支持模块

3.1 PRMTOP 文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "prmtop",
    "scopeName": "source.prmtop",
    "fileTypes": [
        "prmtop"
    ],
    "patterns": [
        {
            "match": "\\%VERSION|\\%FLAG|\\%FORMAT",
            "name": "keyword.prmtop"
        },
        {
            "match": "(?<=\\%FORMAT)\\S+",
            "name": "support.type.format.prmtop"
```

```
        },
        {
            "//": "float",
            "match": "\\s+-?\\d+\\.\\d*",
            "name": "constant.numeric.float.prmtop"
        },
        {
            "match": "\\s+-?\\d+",
            "name": "constant.numeric.integer.prmtop"
        },
        {
            "//": "use constant as itp?",
            "match": "E[+-]\\d\\d",
            "name": "entity.name.function.exponential.prmtop"
        },
        {
            "begin": "ATOM\\_NAME",
            "end": "CHARGE",
            "//": "\\%FORMAT;\\s+\\n",
            "patterns": [
                {
                    "name": "keyword.prmtop",
                    "match": "\\%VERSION|\\%FLAG|\\%FORMAT"
                },
                {
                    "//": "use \\S+ instead of \\b([A-Z0-9a-z+-]+)\\b to include names
with '+-",
                    "match": "(?<=\\%FORMAT)\\S+",
                    "name": "support.type.format.prmtop"
                },
                {
                    "match": "\\S+",
                    "name": "entity.name.tag.atom-name.prmtop"
                }
            ]
        },
        {
            "begin": "ATOMIC\\_NUMBER\\s+\\n",
            "end": "MASS",
            "patterns": [
                {
                    "name": "keyword.prmtop",
                    "match": "\\%VERSION|\\%FLAG|\\%FORMAT"
                },
                {
                    "match": "(?<=\\%FORMAT)\\S+",
                    "name": "support.type.format.prmtop"
                },
                {
                    "match": "\\b\\d+\\b",
                    "name": "constant.numeric.atom-name.prmtop"
                }
            ]
        },
        {
            "begin": "MASS\\s+\\n",
            "end": "ATOM\\_TYPE\\_INDEX",
```

```json
                "name": "string.quoted.prmtop",
                "patterns": [
                    {
                        "name": "keyword.prmtop",
                        "match": "\\%VERSION|\\%FLAG|\\%FORMAT"
                    },
                    {

                        "match": "(?<=\\%FORMAT)\\S+",
                        "name": "support.type.format.prmtop"
                    }
                ]
            },
            {
                "begin": "RESIDUE\\_LABEL\\s+\\n",
                "end": "RESIDUE\\_POINTER",
                "//": "values of fields",
                "patterns": [
                    {
                        "name": "keyword.prmtop",
                        "match": "\\%VERSION|\\%FLAG|\\%FORMAT"
                    },
                    {

                        "match": "(?<=\\%FORMAT)\\S+",
                        "name": "support.type.format.prmtop"
                    },
                    {

                        "//": "todo: according to protein residue type?",
                        "match": "\\S+",
                        "name": "entity.name.type.residue-name.prmtop"
                    }
                ]
            },
            {
                "begin": "AMBER\\_ATOM\\_TYPE\\s+\\n",
                "end": "TREE\\_CHAIN\\_CLASSIFICATION",
                "//": "values of fields",
                "patterns": [
                    {
                        "name": "keyword.prmtop",
                        "match": "\\%VERSION|\\%FLAG|\\%FORMAT"
                    },
                    {

                        "match": "(?<=\\%FORMAT)\\S+",
                        "name": "support.type.format.prmtop"
                    },
                    {

                        "match": "\\S+",
                        "name": "support.type.atom-type.prmtop"
                    }
                ]
            }
        ]
}
```

3.2 Amber 输入文件语法定义

```
{
  "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage.
    json",
  "name": "Amber or AmberTools (like tleap, parmed) input file",
  "scopeName": "source.in",
  "fileTypes": ["in"],
  "patterns": [
    {
      "name": "comment.line.exclamation",
      "match": "!.*$"
    },
    {
      "name": "comment.line.numbersign",
      "match": "#.*$"
    },
    {
      "name": "entity.name.tag.asterisk",
      "match": "^\\s&\\S+"
    },
    {
      "name": "constant.numeric",
      "match": "\\s+-?\\d+\\.\\d+\\s+|\\s+\\d+\\s+"
    },
    {
      "name": "keyword.md",
      "match": "\\b(imin|iwrap|irest|ntx|cut|ntb|ntc|ntf|tempi|temp0|ntt|gamma_ln|ig|ntp|
pres0|taup|barostat|nstlim|dt|ntpr|ntwx|ntwr)\\b"
    },
    {
      "name": "entity.name.function.format",
      "match": "\\b(pdb|mol2|prmtop|inpcrd|frcmod|itp|top|gro|psf|rtf)\\b"
    },
    {
      "name": "keyword.1.tleap",
      "match": "\\b(source|loadmol2|loadpdb|addAtomTypes|bond|combine|(?i)loadamberparams
|(?i)loadAmberPrep|(?i)loadOFF|(?i)addPdbAtomMap|(?i)addPdbResMap)\\b"
    },
    {
      "name": "keyword.2.tleap",
      "match": "\\b(charge|solvatebox|addIons(2|Rand)|savepdb|saveamberparm|quit)\\b"
    },
    {
      "name": "string.quoted.leap",
      "match": "\\b(leaprc.\\S+)\\b"
    },
    {
      "//": "not matching + !",
      "name": "string.other.ions",
      "match": "\\b(Na+|Cl-|K+|Al3+|Mg2+|Ca2+)\\b"
    },
    {
      "name": "keyword.parmed",
      "match": "\\b(chamber|gromber|writeFrcmod|outparm|parmout)\\b"
    },
```

```
    {
      "name": "support.type.variable",
      "match": "^\\S+(?=\\s*=)"
    }
  ]
}
```

3.3 坐标文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "inpcrd",
    "scopeName": "source.inpcrd",
    "fileTypes": [
        "inpcrd"
    ],
    "patterns": [
        {
            "//": "The second line",
            "match": "^\\s\\d+$",
            "name": "string.quoted.number-of-atom.inpcrd"
        },
        {
            "//": "all float numbers",
            "match": "\\d+\\.\\d+",
            "name": "constant.numeric.coordinates.inpcrd"
        }
    ]
}
```

3.4 预处理文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "prepin",
    "scopeName": "source.prepin",
    "fileTypes": [
        "prepin"
    ],
    "patterns": [
        {
            "name": "comment.line.info.prepin",
            "match": "^.*(?i)remark.*$"
        },
        {
            "name": "comment.line.dummy.prepin",
            "match": "DU(MM)?"
        },
        {
            "name": "entity.name.type.prepin",
```

```
            "match": "\\S+(?=\\s*(INT|XYZ))"
        },
        {
            "name": "variable.other.coord-type.prepin",
            "match": "INT|XYZ"
        },
        {
            "match": "CORRECT|CHANGE|OMIT|NOMIT|BEG|ALL|LOOP|IMPROPER|DONE|STOP",
            "name": "keyword.prepin"
        },
        {
            "name": "constant.numeric.atom-number.prepin",
            "match": "(?<=\\s*)\\d+(?=\\s\\s\\S+\\s+\\S+\\s+[EMBS3456])"
        },
        {
            "name": "entity.name.tag.atom-name.prepin",
            "match": "(?<=\\s+)(\\S+\\s+){2}(?=$)|(?<=\\s+)(\\S+\\s+){4}(?=$)|(?<=\\s*\\d
    +\\s\\s)(\\S+)(?=\\s+\\S+\\s+[EMBS3456])"
        },
        {
            "name": "support.type.atom-type.prepin",
            "match": "(?<=\\s*\\d+\\s\\s\\S+\\s+)\\S+(?=\\s+[EMBS3456])"
        },
        {
            "//": "avoid the following three numbers",
            "name": "keyword.tree-symbol.prepin",
            "match": "(?<=\\s+\\d+\\s+\\S+\\s+\\S+\\s+)[EMBS3456](?!\\s+-?\\d*\\.\\d*|\\s
    +-?\\d*\\s+-?\\d*\\.\\d*)"
        },
        {
            "match": "(?<=-?\\d*\\.\\d*)\\s+-?\\d*\\.\\d*(?=$)",
            "name": "string.quoted.charge.prepin"
        },
        {
            "match": "\\s+-?\\d*\\.\\d*",
            "name": "constant.numeric.float.prepin"
        }
    ]
}
```

3.5 力场修改文件语法定义

```
{
  "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage.
    json",
  "name": "frcmod",
  "scopeName": "source.frcmod",
  "fileTypes": [
    "frcmod"
  ],
  "patterns": [
    {
      "match": "MASS|BOND|ANGL(E)?|DIHE(DRAL)?|IMPROPER|IMPR|NONB(ON)?(D)?(ED)?|^CMAP|\\%
    FLAG|END|SOURCE\\d",
      "name": "keyword.frcmod"
```

```
    },
    {
      "//": "We don't want to include the number in begin, so the last \\s+ might be a
    space after an atomtype",
      "//": "I know atomtypes can only be 1/2 characters long",
      "//": "bug: some files like OPLS write values as integers/nothing after point",
      "begin": "^[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*
    \\s+",
      "end": "$",
      "name": "support.type.atom-type.dihedral.frcmod",
      "patterns": [
        {
          "//": "sometimes k = .25, cannot be shown as constant...",
          "//": "sometimes phi = 180",
          "match": "\\s*\\d+(?=\\s+-?\\d*\\.\\d*\\s+(-?\\d+\\.\\d*|\\d+))",
          "name": "variable.other.redundancy.frcmod"
        },
        {
          "match": "(?<=\\s+\\d+\\s+-?\\d*\\.\\d*\\s+(-?\\d+\\.\\d*|\\d+)\\s+)-?\\d+",
          "name": "string.quoted.dihedral-n.frcmod"
        },
        {
          "//": "all previous in this line, correspond to above. sometimes no nonbond,
    only one number",
          "match": "(?<=^[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*-[A-Za-z
    0-9*]+[ ]*(\\s+\\d+)?\\s+-?\\d*\\.\\d*\\s+(-?\\d+\\.\\d*|\\d+)\\s+-?\\d+.?\\d*).*$",
          "name": "comment.line.dihedral-improper.frcmod"
        },
        {
          "match": "\\b(-?\\d*\\.\\d*)\\b|\\b\\d+\\b",
          "name": "constant.numeric.parameter.frcmod"
        }
      ]
    },
    {
      "//": "angle incorrectly matches dihedral/... section",
      "begin": "^[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*\\s+",
      "end": "$",
      "name": "support.type.atom-type.angle.frcmod",
      "patterns": [
        {
          "match": "(?<=^[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*\\s+-?\\d*
    \\.\\d*\\s+-?\\d*\\.\\d*).*$",
          "name": "comment.line.angle.frcmod"
        },
        {
          "match": "\\b(-?\\d*\\.\\d*)\\b|\\b\\d+\\b",
          "name": "constant.numeric.parameter.frcmod"
        }
      ]
    },
    {
      "//": "bond incorrectly matches angle/... section",
      "begin": "^[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*\\s+",
      "end": "$",
      "name": "support.type.atom-type.bond.frcmod",
      "patterns": [
```

```
      {
        "match": "(?<=^[A-Za-z0-9*]+[ ]*-[A-Za-z0-9*]+[ ]*\\s+-?\\d*\\.\\d*\\s+-?\\d*
\\.\\d*).*$",
        "name": "comment.line.angle.frcmod"
      },
      {
        "match": "\\b(-?\\d*\\.\\d*)\\b|\\b\\d+\\b",
        "name": "constant.numeric.parameter.frcmod"
      }
    ]
  },
  {
    "//": "atom incorrectly matches bond/angle/... section or comment",
    "//": "ions may also contain +- but no bond/angle/...",
    "begin": "^\\s*[A-Za-z0-9*+-]+\\s+",
    "end": "$",
    "name": "support.type.atom-type.atom.frcmod",
    "patterns": [
      {
        "match": "\\b(-?\\d*\\.\\d*)\\b",
        "name": "constant.numeric.parameter.frcmod"
      },
      {
        "//": "after float but not starting with number",
        "match": "(?<=-?\\d*\\.\\d*\\s+)[a-zA-Z#!;\\$].*$",
        "name": "comment.line.mass-nonbon.frcmod"
      }
    ]
  },
  {
    "match": "\\s+-?\\d*\\.\\d*",
    "name": "constant.numeric.cmap-float.frcmod"
  },
  {
    "name": "string.quoted.cmap-integer.frcmod",
    "match": "(?<=^\\%FLAG\\s*\\S+\\s*)\\d+(?=\\s*$)"
  }
  ]
}
```

3.6 库文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "tLeap lib file",
    "scopeName": "source.lib",
    "fileTypes": [
        "lib"
    ],
    "patterns": [
        {
            "name": "comment.line.exclamation",
            "match": "(?<=\\!entry\\.[A-Z0-9]+).*$|!!.*$"
        },
```

```json
        {
            "name": "entity.name.type.residue-names.lib",
            "match": "(?<=^\\s*\")\\S+(?=\"\\s*$)"
        },
        {
            "name": "keyword.lib",
            "match": "!entry"
        },
        {
            "name": "entity.name.type.residue-name.lib",
            "match": "(?<=!entry.)[A-Z0-9]+"
        },
        {
            "//": "overwritten by comment",
            "name": "entity.name.tag.type.lib",
            "match": "(?<=!entry.[A-Z0-9]+.unit.)[a-z]+"
        },
        {
            "name": "entity.name.tag.atom-name.lib",
            "match": "(?<=^\\s+\")\\S+(?=\"\\s+\"\\S+\")"
        },
        {
            "name": "support.type.residue-number.lib",
            "match": "(?<=^\\s+\"\\S+\"\\s+\"\\S+\"\\s\\d\\s)-?\\d+(?=\\s\\d+\\s\\d+\\s-?
\\d+\\s-?\\d+\\.\\d*(E-0\\d)?)"
        },
        {
            "name": "entity.name.function.atom-number.lib",
            "match": "(?<=^\\s+\"\\S+\"\\s+\"\\S+\"\\s\\d\\s-?\\d+\\s\\d+\\s)\\d+(?=\\s-?
\\d+\\s-?\\d+\\.\\d*(E-0\\d)?)"
        },
        {
            "name": "entity.name.tag.element-number.lib",
            "match": "(?<=^\\s+\"\\S+\"\\s+\"\\S+\"\\s\\d\\s-?\\d+\\s\\d+\\s\\d+\\s)-?\\d
+(?=\\s-?\\d+\\.\\d*(E-0\\d)?)"
        },
        {
            "name": "string.quoted.charge.lib",
            "match": "(?<=^\\s+\"\\S+\"\\s+\"\\S+\"\\s\\d\\s-?\\d+\\s\\d+\\s\\d+\\s-?\\d+
\\s)(-?\\d+\\.\\d*(E-0\\d)?)"
        },
        {
            "name": "support.type.atom-type.lib",
            "match": "(?<=^\\s+\"\\S+\"\\s+\")\\S+(?=\")"
        },
        {
            "//": "purely number lines. atom number in other places",
            "match": "(?<=^)[\\s\\d+]+(?=\\s*$)",
            "name": "variable.other.atom-number-index.lib"
        },
        {
            "name": "constant.numeric.lib",
            "match": "\\b\\d+\\b|(?<=^\\s*)\\d+|(\\b|\\-)\\d+\\.?\\d*([eE][\\+\\-]?\\d+)
?"
        }
    ]
}
```

3.7 Antechamber 文件语法定义

```json
{
  "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage.
    json",
  "name": "Amber ac file",
  "scopeName": "source.ac",
  "fileTypes": [
    "ac"
  ],
  "patterns": [
    {
      "name": "keyword.ac",
      "match": "\\b(CHARGE|Formula)\\b"
    },
    {
      "match": "(?<=CHARGE|Formula).*$",
      "name": "comment.line.remark.ac"
    },
    {
      "begin": "^ATOM  |^HETATM",
      "end": "$",
      "name": "keyword.atom.ac",
      "patterns": [
        {
          "name": "constant.numeric.atom-number.ac",
          "match": "(?<=^.{6}).{5}"
        },
        {
          "name": "entity.name.tag.atom-name.ac",
          "match": "(?<=^.{12}).{4}"
        },
        {
          "name": "string.unquoted.alternate-location.ac",
          "match": "(?<=^.{16}).{1}"
        },
        {
          "name": "entity.name.type.residue-name.ac",
          "match": "(?<=^.{17}).{4}"
        },
        {
          "name": "support.type.chain-id.ac",
          "match": "(?<=^.{21}).{1}"
        },
        {
          "name": "variable.other.residue-number.ac",
          "match": "(?<=^.{22}).{4}"
        },
        {
          "name": "entity.name.function.code-for-insertions-of-residues.ac",
          "match": "(?<=^.{26}).{1}"
        },
        {
```

```json
        "name": "constant.numeric.x-coordinate.ac",
        "match": "(?<=^.{30}).{8}"
      },
      {
        "name": "constant.numeric.y-coordinate.ac",
        "match": "(?<=^.{38}).{8}"
      },
      {
        "name": "constant.numeric.z-coordinate.ac",
        "match": "(?<=^.{46}).{8}"
      },
      {
        "name": "string.quoted.charge.ac",
        "match": "(?<=^.{55}).{10}"
      },
      {
        "name": "support.type.atom-type.ac",
        "match": "(?<=^.{68}).{6}"
      }
    ]
  },
  {
    "begin": "^BOND",
    "end": "$",
    "name": "keyword.bond.ac",
    "patterns": [
      {
        "name": "string.other.bond-number.acconstant.numeric.atom-number.ac",
        "match": "(?<=^.{6}).{4}"
      },
      {
        "name": "constant.numeric.atom-number1.ac",
        "match": "(?<=^.{10}).{5}"
      },
      {
        "name": "constant.numeric.atom-number2.ac",
        "match": "(?<=^.{15}).{5}"
      },
      {
        "name": "entity.name.function.bond-type.ac",
        "match": "(?<=^.{20}).{5}"
      },
      {
        "name": "entity.name.tag.atom-name1.ac",
        "match": "(?<=^.{26}).{6}"
      },
      {
        "name": "entity.name.tag.atom-name2.ac",
        "match": "(?<=^.{32}).{4}"
      }
    ]
  }
 ]
}
```

3.8 主链文件语法定义

```json
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "mc",
    "scopeName": "source.mc",
    "fileTypes": [
        "mc"
    ],
    "patterns": [
        {
            "match": "HEAD_NAME|TAIL_NAME|PRE_HEAD_TYPE|POST_TAIL_TYPE|CHARGE",
            "name": "keyword.other.mc"
        },
        {
            "match": "OMIT_NAME",
            "name": "string.other.mc"
        },
        {
            "name": "entity.name.tag.atom-name.mc",
            "match": "(?<=NAME\\s*)\\b\\S+\\b"
        },
        {
            "name": "support.type.atomtype.mc",
            "match": "(?<=TYPE\\s*)\\b\\S+\\b"
        },
        {
            "name": "constant.numeric.mc",
            "match": "\\b(-?\\d*\\.\\d*)\\b|\\b\\d+\\b"
        }
    ]
}
```

四、 Gromacs 格式支持模块

4.1 Gromacs 坐标文件语法定义

```json
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "Gromacs mutation file",
    "scopeName": "source.gro",
    "fileTypes": [
        "gro"
    ],
    "patterns": [
        {
      "name": "keyword.timestep",
      "match": "t=.*$"
        },
        {
```

```
                "//": "the first line? not starting with number or space",
                "name": "comment.line.gro",
                "match": "^[^0-9 ].*$"
            },
            {
        "name": "keyword.total_number",
        "match": "^\\d+\\s*$"
    },
            {
                "name": "keyword.other.box-vector.gro",
                "match": "^(\\s+-?\\d+\\.\\d+)+\\s*$"
            },
            {
                "name": "support.type.residue-number.gro",
                "match": "^.{5}"
            },
            {
                "name": "entity.name.type.residue-name.gro",
                "match": "(?<=^.{5}).{5}"
            },
            {
                "name": "entity.name.tag.atom-name.gro",
                "match": "(?<=^.{10}).{5}"
            },
            {
                "//": "(?<=^.{15}).{5}. Use black",
                "name": "constant.numeric.atom-number.gro",
                "match": ""
            },
            {
                "name": "constant.numeric.coordinate.gro",
                "match": "(?<=^.{20}).{8}|(?<=^.{28}).{8}|(?<=^.{36}).{8}"
            },
            {
                "name": "string.quoted.velocity.gro",
                "match": "(?<=^.{44}).{8}|(?<=^.{52}).{8}|(?<=^.{60}).{8}"
            }
        ]
}
```

4.2 原子类型文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "atp",
    "scopeName": "source.atp",
    "fileTypes": [
        "atp"
    ],
    "patterns": [
        {
            "name": "comment.line.semicolon",
            "match": ";.*$"
        },
```

```json
        {
            "name": "constant.numberic.atp",
            "match": "-?\\d+\\.\\d*"
        },
        {
            "//": "all strings made up of capitalized letters and numbers. matching a lot
    of remaining strings...including dihedral n",
            "name": "support.type.atom-type.atp",
            "match": "\\b([A-Za-z0-9_]+)\\b"
        },
        {
            "//": "in special files there are lower case letters",
            "name": "support.type.atomtype.atp",
            "match": "\\b(amber.*)\\b"
        }
    ]
}
```

4.3 原子重命名文件语法定义

```json
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "arn",
    "scopeName": "source.arn",
    "fileTypes": [
        "arn"
    ],
    "patterns": [
        {
            "name": "comment.line.semicolon",
            "match": ";.*$"
        },
        {
            "name": "constant.numberic.arn",
            "match": "-?\\d+\\.\\d*"
        },
        {
            "//": "the word after RESI|PRES",
            "name": "entity.name.type.residue-name.rtf",
            "match": "(?<=^\\s*)\\S+"
        },
        {
            "//": "add (4/5, after two numbers), improper (4)",
            "name": "entity.name.tag.atom-name.tdb",
            "match": "(?<=^\\s*\\S+\\s*)[0-9]*[A-Z]+[A-Z0-9_'*]*"
        },
        {
            "//": "all strings made up of capitalized letters and numbers. matching a lot
    of remaining strings...including dihedral n",
            "name": "support.type.atom-type.arn",
            "match": "(?<=^\\s*\\S+\\s*[0-9]*[A-Z]+[A-Z0-9_'*]*\\s*)[0-9]*[A-Z]+[A-Z
    0-9_'*]*"
        }
    ]
```

```
}
```

4.4 残基类型文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "Gromacs mutation file",
    "scopeName": "source.rtp",
    "fileTypes": [
        "rtp"
    ],
    "patterns": [
        {
            "name": "comment.line.semicolon",
            "match": ";.*$"
        },
        {
            "name": "comment.line.other",
            "match": "torsion.*$|un.*$"
        },
    {
      "//": "section",
      "match": "\\[.*\\]",
      "name": "entity.name.class"
    },
        {
            "//": "atoms section; bonds",
            "name": "entity.name.tag.atom-name.rtp",
            "match": "[0-9]*[A-Z]+[A-Z0-9_']*(?=\\s+[0-9]*[A-Z]+[A-Z0-9_'*]*\\s+-?\\d+\\.
    \\d*)|(\\s+-?\\+?[0-9]*[A-Z]+[A-Z0-9_'\\+]*){2}(?=\\s*($|;))"
        },
        {
            "//": "impropers/dihedrals; cmap",
            "name": "entity.name.tag.atom-name2.rtp",
            "match": "(\\s+-?\\+?[0-9]*[A-Z]+[A-Z0-9_'\\+]*){4}|(-?\\+?[0-9]*[A-Z]+[A-Z
    0-9_'\\+]*\\s+){4}-?\\+?[0-9]*[A-Z]+[A-Z0-9_'\\+]*"
        },
        {
            "//": "after atom name, match any of these strings",
            "name": "support.type.atom-type.rtp",
            "match": "[0-9]*[A-Z]+[A-Z0-9_'*]*(?=\\s+-?\\d+\\.\\d*)"
        },
        {
            "name": "string.quoted.charge.rtp",
            "match": "(?<=[0-9]*[A-Z]+[A-Z0-9_'*]*\\s+)-?\\d+\\.\\d*"
        },
    {
      "//": "number",
      "match": "(\\b|\\-)\\d+\\.?\\d*([eE][\\+\\-]?\\d+)?",
      "name": "constant.language.rtp"
    }
    ]
}
```

4.5 氢数据库文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "Gromacs hdb file",
    "scopeName": "source.hdb",
    "fileTypes": [
        "hdb"
    ],
    "patterns": [
        {
            "name": "comment.line.exclamation",
            "match": ";.*$"
        },
        {
            "name": "support.type.residue-name.hdb",
            "match": "(?<=^)\\S+(?=\\s*\\d+\\s*$)"
        },
        {
            "name": "string.other.count.hdb",
            "match": "(?<=^\\S+\\s*)\\d+(?=\\s*$)"
        },
        {
            "name": "entity.name.tag.atom-name.hdb",
            "match": "(?<=^\\d+\\s+\\d+\\s+)\\S+\\s+\\S+(\\s+\\S+\\s+\\S+)?(\\s+\\S+)?(?=
    \\s*$)"
        },
        {
            "name": "constant.numeric.hdb",
            "match": "\\b\\d+\\b|(?<=^\\s*)\\d+|(\\b|\\-)\\d+\\.?\\d*([eE][\\+\\-]?\\d+)
    ?"
        }
    ]
}
```

4.6 残基到珠映射文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "Gromacs r2b file",
    "scopeName": "source.r2b",
    "fileTypes": [
        "r2b"
    ],
    "patterns": [
        {
            "name": "comment.line.exclamation",
            "match": ";.*$"
        },
        {
            "name": "entity.name.type.residue-name.r2b",
            "match": "\\S+"
```

```
        },
        {
            "name": "constant.numeric.r2b",
            "match": "\\b\\d+\\b|(?<=^\\s*)\\d+|(\\b|\\\-)\\d+\\.?\\d*([eE][\\+\\-]?\\d+)
    ?"
        }
    ]
}
```

4.7 终端数据库文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "Gromacs terminal database",
    "scopeName": "source.tdb",
    "fileTypes": [
        "tdb"
    ],
    "patterns": [
        {
            "name": "comment.line.semicolon",
            "match": ";.*$"
        },
        {
          "//": "section",
          "match": "\\[.*\\]",
          "name": "entity.name.class"
        },
        {
            "//": "deltete (one space); replace (before type mass charge)",
            "name": "entity.name.tag.atom-name.tdb",
            "match": "(?<=^\\s)[0-9]*[A-Z]+[A-Z0-9_']*|[0-9]*[A-Z]+[A-Z0-9_']*(?=\\s
+[0-9]*[A-Z]+[A-Z0-9_'*]*\\s+-?\\d+\\.\\d*)"
        },
        {
            "//": "add (4/5, after two numbers), improper (4)",
            "name": "entity.name.tag.atom-name2.tdb",
            "match": "(\\s*-?\\+?[0-9]*[A-Z]+[A-Z0-9_'\\+]*){4}(\\s*-?\\+?[0-9]*[A-Z]+[A-
Z0-9_'\\+]*)?"
        },
        {
            "//": "after atom name, match any of these strings",
            "name": "support.type.atom-type.tdb",
            "match": "[0-9]*[A-Z]+[A-Z0-9_'*]*(?=\\s+-?\\d+\\.\\d*)"
        },
        {
            "name": "constant.numeric.atomic-mass.tdb",
            "match": "(?<=[0-9]*[A-Z]+[A-Z0-9_'*]*\\s+)-?\\d+\\.\\d*"
        },
        {
            "name": "string.quoted.charge.tdb",
            "match": "(?<=[0-9]*[A-Z]+[A-Z0-9_'*]*\\s+-?\\d+\\.\\d*\\s+)-?\\d+\\.\\d*"
        },
        {
```

```
        "//": "number",
        "match": "(\\b|\\-)\\d+\\.?\\d*([eE][\\+\\-]?\\d+)?",
        "name": "constant.language.tdb"
    }
    ]
}
```

4.8 突变拓扑文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "Gromacs mutation file",
    "scopeName": "source.mtp",
    "fileTypes": [
        "mtp"
    ],
    "patterns": [
        {
            "name": "comment.line.semicolon",
            "match": ";.*$"
        },
        {
            "name": "keyword.mtp",
            "match": "default\\S+"
        },
    {
      "//": "section",
      "match": "\\[.*\\]",
      "name": "entity.name.class"
    },
        {
            "//": "atoms section; impropers/rotations/etc; the last word in rotation is
    not matched? sometimes -XX is connected to the previous name",
            "name": "entity.name.tag.atom-name.mtp",
            "match": "[0-9]*[A-Z]+[A-Z0-9_']*(?=\\s+[0-9]*[A-Z]+[A-Z0-9_*]*)|\\S+.gone\\S
    *|(\\s+-?\\+?[0-9]*[A-Z]+[A-Z0-9_-]*){4}"
        },
        {
            "//": "after atom name, match any of these strings",
            "name": "support.type.atom-type.mtp",
            "match": "[0-9]*[A-Z]+[A-Z0-9_'*]*"
        },
        {
            "name": "string.quoted.charge.mtp",
            "match": "(?<=[0-9]*[A-Z]+[A-Z0-9_*]*\\s+)-?\\d+\\.\\d*"
        },
    {
      "//": "number",
      "match": "(\\b|\\-)\\d+\\.?\\d*([eE][\\+\\-]?\\d+)?",
      "name": "constant.language.mtp"
    }
    ]
}
```

4.9 虚拟位点定义文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "Gromacs virtual site definition file",
    "scopeName": "source.vsd",
    "fileTypes": [
        "vsd"
    ],
    "patterns": [
        {
            "name": "comment.line.semicolon",
            "match": ";.*$"
        },
        {
            "name": "keyword.vsd",
            "match": "planar"
        },
    {
      "//": "section",
      "match": "\\[.*\\]",
      "name": "entity.name.class"
    },
        {
            "//": "aromatic section",
            "name": "entity.name.tag.atom-name.vsd",
            "match": "[0-9]*[A-Z]+[A-Z0-9_']*(?=.*\\s+\\d+\\.?\\d*)"
        },
        {
            "//": "after atom name, match any of these strings",
            "name": "support.type.atom-type.vsd",
            "match": "[0-9]*[a-zA-Z]+[A-Z0-9_'*]*"
        },
    {
      "//": "number",
      "match": "(\\b|\\-)\\d+\\.?\\d*([eE][\\+\\-]?\\d+)?",
      "name": "constant.language.vsd"
    }
    ]
}
```

五、 小分子格式支持模块

5.1 MOL2 文件语法定义

```
{
    "patterns": [
        {
            "include": "#structure"
        }
    ],
```

```json
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "scopeName": "source.mol2",
    "name": "mol2",
    "repository": {
        "structure": {
            "patterns": [
                {
                    "begin": "^#",
                    "end": "$",
                    "name": "comment.line"
                },
                {
                    "//": "not matching",
                    "match": "(?<=(MOLECULE\\s*\\n))\\S+",
                    "name": "entity.name.function.molname.mol2"
                },
                {
                    "begin": "^@",
                    "end": "$",
                    "name": "string.quoted.orange",
                    "//": "values of fields",
                    "patterns": [
                        {
                            "match": "(?<=(>))[A-Za-z\\_]+",
                            "name": "keyword.mol2"
                        }
                    ]
                },
                {
                    "//": "in the follwing big section, the first number is matched and
    screened...maybe use a new color",
                    "match": "\\d+(?=\\s+\\d+\\s+\\d+\\s+[0-9a-z]+$)",
                    "name": "string.quoted.bond-number.mol2"
                },
                {
                    "//": "all lines beginning with a number, including bond section and
    mol info",
                    "begin": "\\d+",
                    "end": "$",
                    "name": "constant.character.coordinates.mol2",
                    "patterns": [
                        {
                            "//": "the first number is not here at all, but we might not
    need to highlight it.",
                            "match": "(?<=\\s*)\\d+(?=\\s[A-Z]+[A-Za-z0-9_'\\.]*)",
                            "name": "constant.numeric.atom-number.mol2"
                        },
                        {
                            "match": "(?<=\\d+\\s+)[A-Z]+[A-Za-z0-9_'\\.]*+(?=(\\s+(-?\\d
    +\\.\\d*)\\s+(-?\\d+\\.\\d*)\\s+(-?\\d+\\.\\d*)))",
                            "name": "variable.other.atom-name.mol2"
                        },
                        {
                            "//": "No ((?=(\\s+\\d+\\s+[a-zA-Z0-9\\_]+\\s+[\\.0-9\\-]+[
    \\s\\*]*$))|(?=[\\s]*$)), sometimes no charge",
                            "match": "(?<=(-?\\d+\\.\\d+\\s+-?\\d+\\.\\d+\\s+-?\\d+\\.\\d
```

```
                +\\s+))[A-Z]+[A-Za-z0-9_\\.]*",
                            "name": "support.type.atom-type.mol2"
                    },
                    {
                            "match": "(?<=[A-Z]+[A-Za-z0-9_\\.]*\\s+)\\d+(?=\\s+[A-Z]+[A-
                Z0-9_'\\+]*)",
                            "name": "variable.other.residue-number.mol2"
                    },
                    {
                            "//": "(?=(\\s+(-?\\d+\\.\\d*)+\\s*$)) charge is optional",
                            "match": "(?<=\\s+\\d+\\s+)[A-Z]+[A-Z0-9_'\\+]*",
                            "name": "entity.name.type.residue-name.mol2"
                    },
                    {
                            "match": "(?<=\\s+\\d+\\s+[a-zA-Z0-9\\_]+\\s+)(-?\\d+\\.\\d+)
                (?<=$)",
                            "name": "string.quoted.charge.mol2"
                    },
                    {
                            "match": "(?<=^\\s*\\d+\\s+\\d+\\s+\\d+\\s+)[0-9a-z]+(?=\\s
                *$)",
                            "name": "entity.name.function.bond-type.mol2"
                    }
                ]
            },
            {
                "//": "^[a-zA-Z_']+, too blue!",
                "match": "",
                "name": "support.function.field"
            }
        ]
    }
}
}
```

5.2 SDF 文件语法定义

```
{
    "name": "sdf",
    "scopeName": "source.sdf",
    "fileTypes": ["sdf"],
    "patterns": [
        {
            "//": "float starting from column 1, no other",
            "match": "^\\d+(\\.\\d+)$",
            "name": "string.quoted.double.sdf"
        },
        {
            "//": "string starting from column 1, no other",
            "match": "^(?!\\${4})\\S+$",
            "name": "string.quoted.double.sdf"
        },
        {
            "//": "string starting from column very later, no other",
            "match": "^\\s+\\S+$",
```

```
            "name": "support.type.3d.sdf"
        },
        {
            "match": "\\bV2000\\b",
            "name": "keyword.other.sdf"
        },
        {
            "//": "special string properties, 3/4 strings",
            "match": "^((\\S+\\s+){2})+\\S+$|^((\\S+\\s+){3})+\\S+$",
            "name": "string.quoted.double.sdf"
        },
        {
            "//": "END and $$$$",
            "match": "^M\\s+END$|^\\${4}$",
            "name": "support.type.end.sdf"
        },
        {
            "match": "^>\\s<.+>$",
            "name": "keyword.other.sdf"
        },
        {
            "match": "\\s+(-?\\d+\\.\\d+)\\s+(-?\\d+\\.\\d+)\\s+(-?\\d+\\.\\d+)",
            "name": "constant.numberic.atom-coordinate.sdf"
        },

        {
            "match": "\\b([A-Za-z]+)\\b",
            "name": "string.quoted.element-symbol.sdf"
        },
        {
            "match": "^\\s+\\d+\\s+\\d+",
            "name": "entity.name.tag.bond.sdf"
        },
        {
            "match": "(?<=^\\s+\\d+\\s+\\d+)\\s+\\d+",
            "name": "string.quoted.bond-order.sdf"
        }
    ]
}
```

六、 专业工具支持模块

6.1 PLUMED 输入文件语法定义

```
{
    "$schema": "https://raw.githubusercontent.com/martinring/tmlanguage/master/tmlanguage
    .json",
    "name": "plumed",
    "scopeName": "source.plumed",
    "fileTypes": [
        "plumed"
    ],
    "patterns": [
```

```
        {
            "name": "comment.line.plumed",
            "match": "^\\#.*$"
        },
        {
            "name": "string.quoted.action.plumed",
            "match": "\\b(GROUP|MOLINFO|PRINT|ANGLE|TORSION|DISTANCE|COM|GYRATION|VOLUME)
    \\b"
        },
        {
            "name": "string.quoted.bias.plumed",
            "match": "\\b(LOWER_WALLS|UPPER_WALLS)\\b"
        },
        {
            "name": "keyword.common.plumed",
            "match": "\\b(ATOMS|ARG|FILE|STRIDE|NDX_FILE|NDX_GROUP|RESTRAINT|METAD)\\b"
        },
        {
            "name": "support.type.argument.plumed",
            "match": "\\b(AT|KAPPA|EXP|EPS|OFFSET)\\b"
        },
        {
            "name": "variable.name.argument.plumed",
            "match": "^\\S+(?=:)"
        },
        {
            "name": "variable.name.plumed",
            "match": "(?<=(ARG|ATOMS|FILE)=)\\S+(?=\\s+)"
        },
        {
            "name": "constant.numeric.plumed",
            "match": "-?\\d+\\.?\\d*"
        }
    ]
}
```