MATCH Users' Manual

Joseph D. Yesselman and Charles L. Brooks III

10/10/2011

---

1 Installation

```
tar xvzf MATCH_RELEASE.tar.gz
```

Will create a folder called MATCH_RELEASE/ unless otherwise specified

First step required from installation is to set 2 environment variables, if one is using bash this can be accomplished using the following commands

```
export PerlChemistry=insert_your_path_here/MATCH_RELEASE/PerlChemistry
export MATCH=insert_your_path_here/MATCH_RELEASE/MATCH
```

if you are using cshell/tcsh, this can be checked by typing "echo $SHELL"use instead

```
setenv PerlChemistry insert_your_path_here/MATCH_RELEASE/PerlChemistry
setenv MATCH insert_your_path_here/MATCH_RELEASE/MATCH
```

please be advised that MATCH was developed using Perl 5.10, entering the command "perl –v"will give you the version information for Perl.

IIMPORTANT: if you do not add the MATCH_RELEASE/scripts/ to your $PATH as this is the location of MATCH.pl the wrapper script used to type and parameterize new molecules in all the examples in which we refer to MATCH.pl you will have have to type out the ENTIRE PATH of where MATCH.pl is located in your file system.

2 Short Overview of MATCH Scripts

2.1 MATCH.pl

MATCH.pl is a wrapper script that performs atom type assignment, atom charging and generates inter and intra molecular parameters for a given novel molecule

As default the Charmm General Forcefield will be used for charging and parameterization of novel molecules, this can be explicitly declared by adding -forcefield top_all36_cgenff, see examples. We are also in the process of validating a new version of the Charmm General Forcefield, this can be used instead by adding -forcefield top_all36_cgenff_new. Later versions of MATCH will use this by default but for now it is recommended to use top_all36_cgenff.

```
MATCH.pl [options] filename
```

2.1.1 Options

−AppendingTopologyFilePath	The topology file that one will be using to simulate the receptor, this is important the default value for this is top_all22_prot.inp if one plans to use any forcefield other than this one it has to be specified here. Failure to do this will assign the ligand atom types to MASS numbers that already exist and will lead to an incredible amount of errors.
−AtomicProtonationStates	Used to force atoms into a certain protonation state, this is optional, an example would be NZ=1 this would force the atom named NZ to have a protonation stat of +1
−BondLengthFilePath	The location of the path that contains acceptable bond lengths to allow for atoms without bonding information to be connected see MATCH/resources/BondLengths.dat This option exists so users do not have to edit this file as it has been shown to handle most small molecules.
−CreatePdb	All file formats other than pdbs are not supported by CHARMM, using this option followed by a filename will output the coordinates in CHARMM pdb format. This is also advised if the input pdb does not have unique atom names as this is also a requirement of MATCH/CHARMM.
−ExitifNotInitiated	If this flag is set to 0 is set it will force MATCH to continue if a molecule fails basic setup such as bond assignment / bond order assignment, useful if batch processing molecules (default is 1)
−ExitifNotTyped	If this flag is set to 0 it will force MATCH to continue if a molecule fails to have all of its atoms typed, useful if batch processing molecules (default is 1)
−ExitifNotCharged	If this flag is set to 0 it will force MATCH to continue if a molecule fails to have all of its atoms charged, useful if batch processing molcules (default is 1)
−Forcefield	The force field that the given molecule will be typed/parameterized the options are top_all22_prot, top_all27_na, top_all35_carb, top_all35_ether, top_all36_cgenff, top_all36_cgenff_new, top_all36_lipid
−SubstituteIncrements	If this flag is set to 0 only increments that have been derived directly from a force field will be used (default is 1)
−UsingRefiningIncrements	If this flag is set to 0 no refining increments (secondary increments) will be applied (default is 1)

The following options are to use custom MATCH libraries not already contained in the MATCH directory

−IncrementFilePath	The location of the increment file path, this file holds the accepted increment learned from a given force field
−ImproperFilePath	The location of the improper file path, this file contains the instructions of how to apply improper dihedral angles to a given force field
−ParameterFilePath	The location of the a force field corresponding parameter file. This is where MATCH looks for which bond, angle, dihedral, improper, and nonbonded parameters to use for novel molecules
−RelationMatrixFilePath	The location of the relation matrix file path, this file stores how related one type is to another this file can be generated by running GenerateTypeRelationMatrix.pl on the topology file of interest.
−RefiningIncrementsFilePath	The location of the refining increments file path, secondary level increments are stored here with their corresponding super smiles string to describe what environment they will applied over the default increment.
−ShortenTypeFilePath	If the atom type names being used are longer than CHARMM can handle, this file holds shortened versions of each type name so it can be compatible with CHARMM.
−TypesFilePath	The location of the types file, it contains the super smiles definitions of all types of a given force field

2.1.2 Acceptable Input FileType Format

For each supported format, an example will be given of the ideal format that if one is having trouble reading in a file can use as an example. Keep in mind MATCH is very versatile this should be ONLY considered if you are getting an error from reading in a file.

Protein DataBank (PDB):

Atom Declaration

```
ATOM      1  C    UNK    0       0.000   0.000   0.000  1.00  0.00      TEST
```

Bond Declaration

```
CONECT 1 2
```

Molfile (MOL/SDF):

Atom Declaration

```
    1.9050   -0.7932    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
```

Bond Declaration

```
  2  1  1  0  0  0  0
```

Molfile (MOL2):

Atom Declaration

```
   1 N1       -0.7256    -0.6895     0.1909 N.3      1  R0_ -1.0098
```

Bond Declaration

```
   1    1    2 1
```

2.1.3 Understanding the Output

Using a pdb of alanine we can see the text output of MATCH.pl

```
MATCH.pl -forcefield top_all36_cgenff ala.pdb
```

Below is the output upon typing/parameterizing alanine with the CHARMM General Force Field (CGENFF). The output can broken into 3 distinct parts. First a list of atoms given by name with their state (element . numofbonds), assigned type and the super smiles string of the type. This information can be useful for debugging an error.

The next section outlines which increments were used for each bond, and what bond type was assigned to the bond.

```
N       N.4     NG3P3   N.4(H.1)(H.1)(H.1)
HT1     H.1     HGP2    H.1(N.4)
HT2     H.1     HGP2    H.1(N.4)
HT3     H.1     HGP2    H.1(N.4)
CA      C.4     CG314   C.4(H.1)(N.4)
HA      H.1     HGA1    H.1(C.4)
CB      C.4     CG331   C.4(H.1)(H.1)(H.1)
HB1     H.1     HGA3    H.1(C.4(H.1)(H.1))
HB2     H.1     HGA3    H.1(C.4(H.1)(H.1))
HB3     H.1     HGA3    H.1(C.4(H.1)(H.1))
C       C.3     CG2O3   C.3(O.1)(O.1)
OT1     O.1     OG2D2   O.1(!C.3(O.1))
OT2     O.1     OG2D2   O.1(!C.3(O.1))
```

```
HT1     HGP2    N       NG3P3   1        0.33000 -  -0.08000
HT2     HGP2    N       NG3P3   1        0.33000 -  -0.08000
HT3     HGP2    N       NG3P3   1        0.33000 -  -0.08000
CA      CG314   N       NG3P3   1        0.31000 -  -0.06000
CA      CG314   HA      HGA1    1       -0.10000 -   0.10000
CA      CG314   CB      CG331   1        0.00000 -   0.00000
C       CG2O3   CA      CG314   1        0.00000 -   0.00000
CB      CG331   HB1     HGA3    1       -0.09000 -   0.09000
CB      CG331   HB2     HGA3    1       -0.09000 -   0.09000
CB      CG331   HB3     HGA3    1       -0.09000 -   0.09000
C       CG2O3   OT1     OG2D2   2        0.17000 -  -0.67000
C       CG2O3   OT2     OG2D2   1        0.17000 -  -0.67000
ala Success!
```

2.1.4 Output files

Using a pdb of alanine we can see the files outputted of MATCH.pl

```
MATCH.pl -forcefield top_all36_cgenff ala.pdb
```

ala.rtf

```
 * Charmm RTF built by MATCH
*
  22     0
MASS   125 HGA1  1.008000 H
MASS   127 HGP2  1.008000 H
MASS   128 N3P3  14.00700 N
MASS   122 C2O3  12.01100 C
MASS   124 C331  12.01100 C
MASS   126 HGA3  1.008000 H
MASS   129 O2D2  15.99900 O
MASS   123 C314  12.01100 C

AUTO ANGLES DIHE

RESI  ALA   -0.000000
GROUP
ATOM N    N3P3  -0.300000
ATOM HT1  HGP2   0.330000
ATOM HT2  HGP2   0.330000
ATOM HT3  HGP2   0.330000
ATOM CA   C314   0.210000
ATOM HA   HGA1   0.100000
ATOM CB   C331  -0.270000
```

```
ATOM HB1  HGA3   0.090000
ATOM HB2  HGA3   0.090000
ATOM HB3  HGA3   0.090000
ATOM C    C2O3   0.340000
ATOM OT1  O2D2  -0.670000
ATOM OT2  O2D2  -0.670000
BOND HT1  N
BOND HT2  N
BOND HT3  N
BOND CA   N
BOND CA   HA
BOND CA   CB
BOND C    CA
BOND CB   HB1
BOND CB   HB2
BOND CB   HB3
BOND C    OT1
BOND C    OT2
IMPR C    CA   OT1  OT2
PATCH FIRST NONE LAST NONE
```

ala.prm

```
* prm file built by MATCH
*

BONDS
HGP2   N3P3   403.00     1.0400
C314   N3P3   200.00     1.4800
C314   HGA1   309.00     1.1110
C314   C331   222.50     1.5380
C2O3   C314   200.00     1.5220
C331   HGA3   322.00     1.1110
C2O3   O2D2   525.00     1.2600

ANGLES
HGP2   N3P3   HGP2   44.00      109.50
HGP2   N3P3   C314   30.00      109.50
HGA1   C314   N3P3   51.50      107.50
C331   C314   N3P3   67.70      110.00
C2O3   C314   N3P3   43.70      110.00
C331   C314   HGA1   34.50      110.10
C2O3   C314   HGA1   50.00      109.50
C2O3   C314   C331   52.00      108.00
C314   C331   HGA3   33.43      110.10
```

```
O2D2   C2O3   C314   40.00      116.00
HGA3   C331   HGA3   35.50      108.40
O2D2   C2O3   O2D2   100.00     128.00

DIHEDRALS
HGP2   N3P3   C314   HGA1   0.1000     3      0.00
HGP2   N3P3   C314   C331   0.1000     3      0.00
HGP2   N3P3   C314   C2O3   0.1000     3      0.00
HGA3   C331   C314   N3P3   0.2000     3      0.00
O2D2   C2O3   C314   N3P3   3.2000     2      180.00
HGA3   C331   C314   HGA1   0.1950     3      0.00
O2D2   C2O3   C314   HGA1   0.0500     6      180.00
O2D2   C2O3   C314   C331   0.0500     6      180.00
C2O3   C314   C331   HGA3   0.1600     3      0.00

IMPROPER
C2O3   X      X      O2D2   96.0000    0      0.00

NONBONDED nbxmod  5 atom cdiel shift vatom vdistance vswitch -
cutnb 14.0 ctofnb 12.0 ctonnb 10.0 eps 1.0 e14fac 1.0 wmin 1.5
HGP2   0.0000     -0.0460    0.2245
N3P3   0.0000     -0.2000    1.8500
C314   0.0000     -0.0310    2.1650
HGA1   0.0000     -0.0450    1.3400
C331   0.0000     -0.0780    2.0500
C2O3   0.0000     -0.0700    2.0000
HGA3   0.0000     -0.0240    1.3400
O2D2   0.0000     -0.1200    1.7000
```

2.1.5 Examples CHARMM Input File

To load the prm and rtf files of just the molecule processed by MATCH:

```
* title
*

READ RTF CARD NAME ala.rtf

READ PARAM CARD NAME ala.prm

READ SEQU PDB NAME ala.pdb

GENERATE TEST FIRST NONE LAST NONE SETUP

READ COOR PDB NAME ala.pdb RESID
```

To load the molecule with a receptor

```
* title
*

READ RTF CARD NAME top_all22_prot.inp

READ RTF APPEND CARD NAME test.rtf

READ PARAM CARD NAME par_all22_prot.inp

READ PARAM APPEND CARD NAME test.prm

READ SEQU PDB NAME test.pdb

GENERATE TEST FIRST NONE LAST NONE SETUP

READ COOR PDB NAME test.pdb RESID
```

2.1.6 Examples

```
MATCH.pl -forcefield top_all36_cgenff test.pdb
```

Create a topology and parameter file for molecule in test.pdb using the CGENFF force field, will be named test.rtf and test.prm

```
MATCH.pl -forcefield top_all22_prot -CreatePdb test test.mol
```

Generates a topology and parameter file for the molecule within test.mol, in addition it outputs CHARMM formatted pdb named test.pdb

```
MATCH.pl -forcefield top_all35_carb test1.mol2 test2.mol2 test3.mol2
```

Generates a topology file for the molecule found in test1, test2, and test3. Only one parameter file will be generated that will contain all of the parameters for test1, test2, and test3.

```
MATCH.pl -forcefield top_all36_cgenff -AppendingTopologyFilePath top_all27_prot_na.rtf
test1.mol
```

Generates a topology file and parameter files for the molecule found in test1.mol, importantly the MASS statements found in the top of test1.rtf now do no conflict with the top_all27_prot_na.rtf

2.1.7 Common Errors

Can't locate BaseObject.pm in @INC (@INC contains: ...

you did not add the correct location for the $PerlChemistry and $MATCH environmental variables please refer to the installation section

____ has too many bonds for its element, structure is warped

The atom mentioned has been predicted to bond to too many atoms. Either you can supply the correct bonding using a MOL/MOL2/SDF file or change the Bondlength file to be more strict using -BondLengthFilePath argument at command line.

Could Not Type Atom ____ With State ____

It is not possible to type this atom, first thing to check is the atom state reported. Which is in the format Atom Element . Number of bonds. If either the atom element is wrong or the number of bonds is wrong that is the likely reason why typing failed. The atom element being wrong is rather rare something CL atoms are mistaken as C due to the atom name. If the number of bonds is wrong then follow a similar procedure as the following error. If both are correct then it is most likely that the forcefield does not cover the given chemical environment. Where there is not much that can be done.

The increment between atoms ____ (____) and ____ (____) does not exist and Increment Substitution failed to yield an acceptable result

First check the types which are the atoms are assigned as they are within the parentheses next to the atom information. Investigating whether the atom type makes sense for the given chemical environment is the first step in debugging this error. If the types are okay then most likely there is not much that can be done other than to manually assign the increment by looking at the existing increments of the .incr file of the force field of interest.