

# A Tutorial for TIPS I, and How to Assemble Paths

Dylan Goldsborough

8<sup>th</sup> of July 2016

# Table of contents

- 1 Introduction
- 2 Example: alanine dipeptide
- 3 Processing TIPS1 output
- 4 Example: DNA baseroll
- 5 Appendices
- 6 Summary

# Introduction

Who am I?

- Dylan Goldsborough
- Student at the computational science master
- Attended the course on biomolecular simulation
- Enjoyed figuring out TIPSII in class
- Did a project to write a tutorial that introduces TIPSII over the past months!

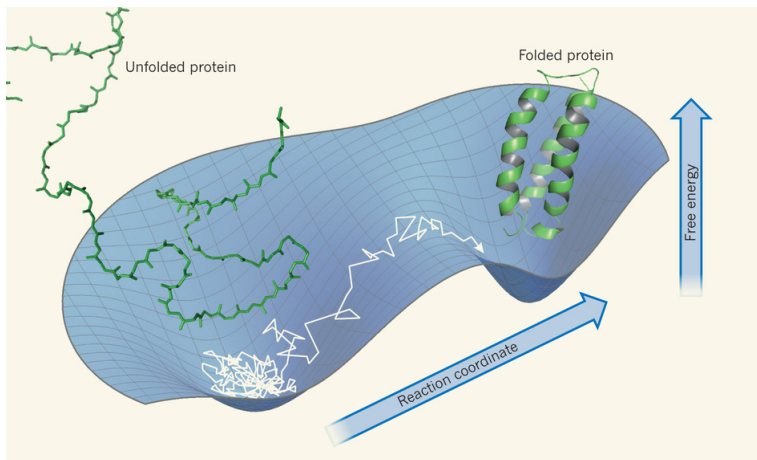
# Outline

In this presentation I will:

- Give a brief recap of TPS
- Outline my contribution
- Discuss both the examples the tutorial offers
- Discuss the additional documentation

# What is TPS?

**Find trajectories of rare transitions:**



(Chung and Eaton, 2013)

# What is TPS?

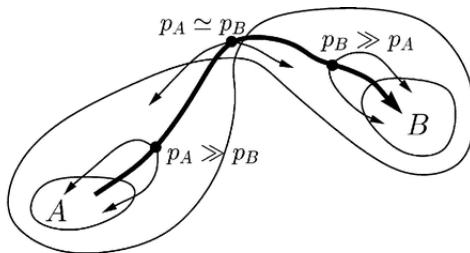
## Algorithm:

- Take existing path
- Choose random time slice  $t$
- Change momenta slightly at  $t$
- Integrate forward or backward in time to create new path
- Accept if state  $A$  or  $B$  is reached, otherwise reject and retain old path

**Important:** we assume that both states are stable, and that we have a transition trajectory ready!

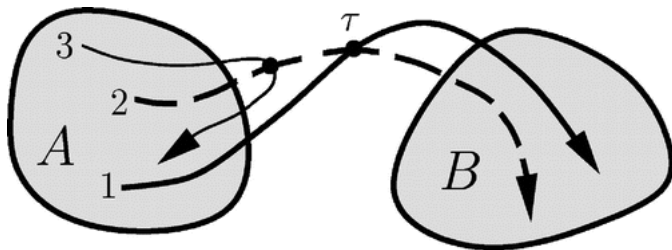
# What is TPS?

## Committer probability:



# What is TPS?

**Shooting move:**



Accepted if state A or B is reached.  
(Bolhuis, 2002)



# What is TIPSİ?

TIPSİ is a script by Tsjerk Wassenaar, which:

- Adaptation of a Perl script by Jarek Juraszek
- Written in Python
- It relies on GROMACS version 4.5.4 (a molecular dynamics engine)
- Does random shooting moves forward and backward
- Reverses time for backward shooting
- A "molecular calculator"

# Contributions

I made several minor contributions to this project:

- Bash implementation to assemble paths
- Python script to analyze the paths
- Tutorial with a Python tool

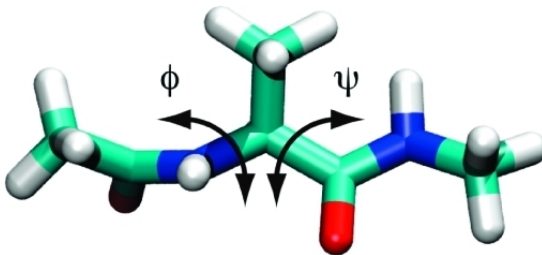
Resulted in some surprises that had to be dealt with...

# Alanine Dipeptide

Our first example is **alanine dipeptide**:

- Extremely fast to run
- Very simple to understand, with simple order parameters (dihedrals)
- Has a nice transition between two distinct states
- Common example to use

# Alanine Dipeptide



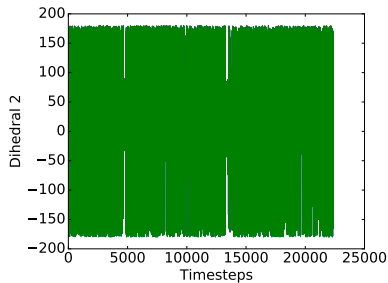
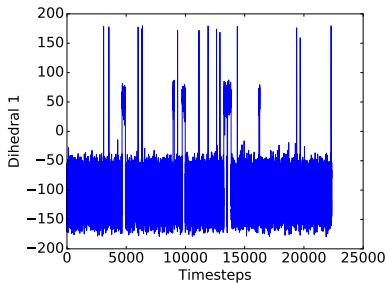
# Preparing for TIPSII

In this example we do everything from scratch:

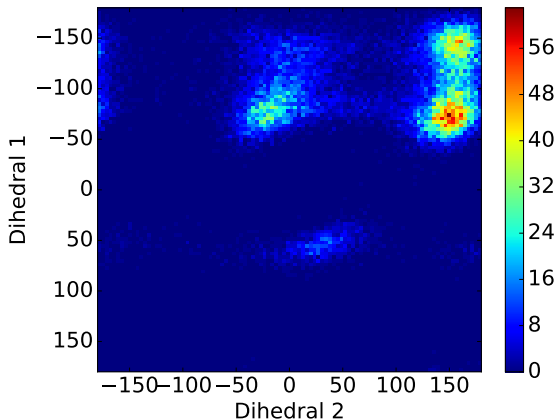
- Start with pdb-file (defines the structure) and MD settings
- We prepare make a periodic system with a solvent
- We do energy minimization followed by a constrained MD run
- We do runs at room temperature and high temperature

We look at the order parameters in the run using a Python script and GROMACS commands.

# Defining the stable states



# Defining the stable states



# Running TIPSİ

```
maxframes    = 2000

par dh1      = dihdeg(frame$Dihedral1)
par dh2      = dihdeg(frame$Dihedral2)

state A = (-150 < dh1 & dh1 < -50
           & 120 < dh2 & dh2 < 180)
state B = (-100 < dh1 & dh1 < -50
           & -50 < dh2 & dh2 < 20)

interface I = (!A) & (!B)
```



# Assembling paths

TIPSİ only saves the shooting moves to save space, so to view and analyze the transitions we need to assemble them.

## The problem:

- TIPSİ outputs shooting moves only
- Instructions on how the paths are made in dat-files
- Backward shooting moves are the wrong way around
- Negative timestamps

**Result:** assembling the paths is a minor nightmare...

# TIPSİ output

## Example of Tipsi output:

```
DIR: tipsi-tutorial/output/jobname/DATA/9/1
```

9-1-BW.cpt	9-1-BW.out	9-1-BW.xtc	DONE
9-1-BW.dat	9-1-BW.top	9-1.dat	md-prod.mdp
9-1-BW.edr	9-1-BW.tpr	ACCEPTED	PARENT
9-1-BW.err	9-1-BW.trr	CMD	parent.dat

# TIPSİ output

REGEX: '[A-I]\s+(\S+)\s+[F-T][a-z]+\s+(\S+)\s+'

state	time	stop	file
A	-1325.0	True	../.. /9/1/9-1-BW.trr
I	-1305.0	False	../.. /9/1/9-1-BW.trr
I	-1285.0	False	../.. /9/1/9-1-BW.trr
I	-1265.0	False	../.. /9/1/9-1-BW.trr
...			
I	465.0	False	../.. /6/4/6-4-FW.trr
I	485.0	False	../.. /6/4/6-4-FW.trr
I	505.0	False	../.. /6/4/6-4-FW.trr
B	525.0	True	../.. /6/4/6-4-FW.trr

# Assembling paths

## The solution:

- Bashscripts that searches for directories with ACCEPTED-file
- Scans dat-file using regular expressions
- Overwrites timestamps
- Creates each path by dumping and appending single frames
- Stores some metainformation in a csv-file

Takes a while to dump frames for long trajectories...

# Metainformation paths

We can do some nice things with the metainformation, I wrote a script that:

- Finds the average path length
- Finds the number of decorrelated groups of paths
- Ratio of FW:BW shooting moves
- Draws a tree of all shooting moves

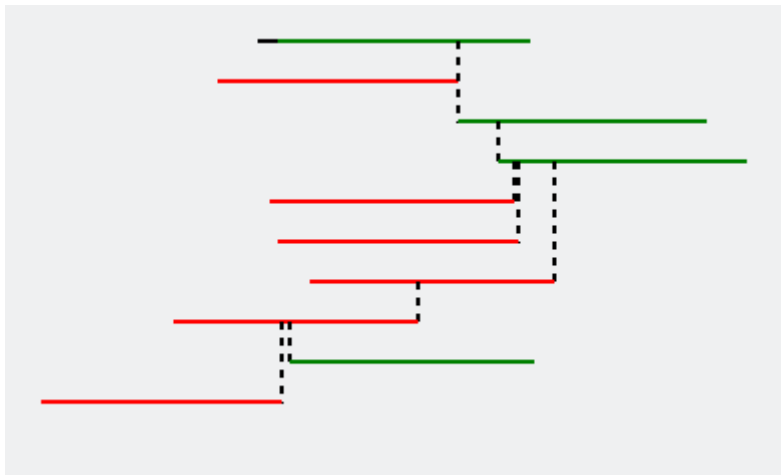
# Analyzing TIPSII

I appended the bashscript to the end of the job, so that it puts all paths and the metadata in the output/DATA directory, now we can take a look at the trajectory in VMD!

We find that:

- Average length is 136.4 frames
- FW/BW ratio is 0.4
- Number of decorrelated groups of paths is 2

# Analyzing TIPSII



We visualize path 9 in VMD.

# DNA baseroll

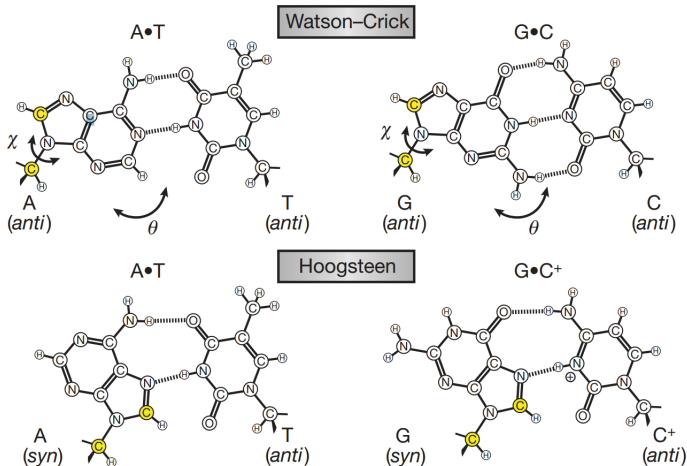
Second example is the DNA baseroll:

- Project by Jocelyne Vreede and David Swenson
- Concerns the transition from the WC- to the HG-pairing
- An actual rare event (microsecond range), unlike example 1

Metadynamics simulation has been done, so there is a trajectory available that contains the transition!



# DNA baseroll

**a**

# Preparing for TIPSİ

In this example we start with a trajectory and state definitions:

- We import the custom topology
- We create a `tpr`-file (containing the simulation settings) to suit our MD-needs
- We set up a parameter file that include the state definitions (H-bonds)
- We run TIPSİ with this `tpr` and the provided trajectory

We look at the order parameters in the run using a Python script and GROMACS commands.

# Analyzing TIPS I

We find that:

- Average length is 723.88 frames
- FW/BW ratio is 0.33
- Number of decorrelated groups of paths is 1

# Analyzing TIPSII

Not very balanced tree:



We visualize path 9 in VMD.

# Cheatsheets

As an appendix, I included:

- A Linux cheatsheet, for students who are unfamiliar
- A GROMACS cheatsheet (which I caught myself use often too)
- A TIPSİ cheatsheet

The last addresses the documentation problem with TIPSİ.

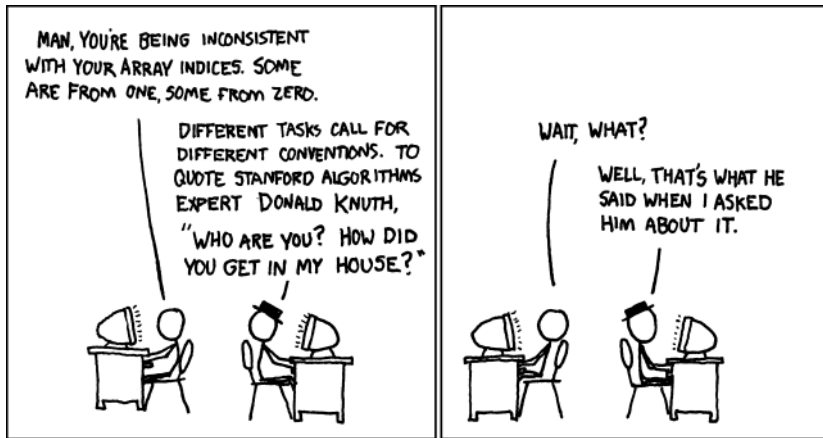
# Cheatsheets

TIPSII cheatsheet includes:

- How to set up a parameter file
- All calculator options
- Several ways to define groups of atoms
  - Numpy arrays
  - Groups from GROMACS ndx-files (defines groups of atoms by their number)

There are **indexing problems**...

# Relevant XKCD



# Running TIPSİ

Calculator options are:

- `com`: center of mass
- `dist`: distance (several options)
- `angle`: angle of 3 atoms
- `dihrad/dihdeg`: dihedral in rads/degrees
- `rgyr`: radius of gyration, incompatible with GROMACS
- `rmsd`: root-mean-square-deviation
- `hbonds`: number of hydrogen bonds, specific or non-specific



# Summary

- I wrote a tutorial for students/people interested in TIPSII with a fast and uninteresting example, and a more computationally demanding but actually interesting example
- Added a bash script that finishes the output to make it fit for further analysis
- Added a basic script that helps understand the shooting moves made
- Compiled all calculator options, and how to set up a parameter file for TIPSII

# Future changes?

There might be some things to change down the line:

- Include OPS as an analysis tool (no time now, and still being developed)
- Append to the possibilities in the parameter file (non-calculator options)
- Any other changes...

The project is on GIT and should be public:

<https://github.com/dgoldsb/tipsitutorial.git>