Interactive Molecular Dynamics (IMDv3) in Practice: MD Packages and Performance

Amruthesh Thirumalaiswamy

Postdoctoral Researcher
School of Molecular Sciences
Arizona State University

MDAnalysis UGM 2025

Outline

This talk will cover how **IMDv3 works in practice**: how IMDv3 is implemented in various Simulation (MD) engines, how we can use it while running simulations, and how well it performs with physical systems of scientific interest.

1. IMDv3 implementation in MD engines
2. IMDv3 usage with GROMACS, LAMMPS and NAMD
3. Benchmarking IMDv3 modified simulations
4. Comparing streaming with file I/O

IMDv3 Functionality in various MD Engines

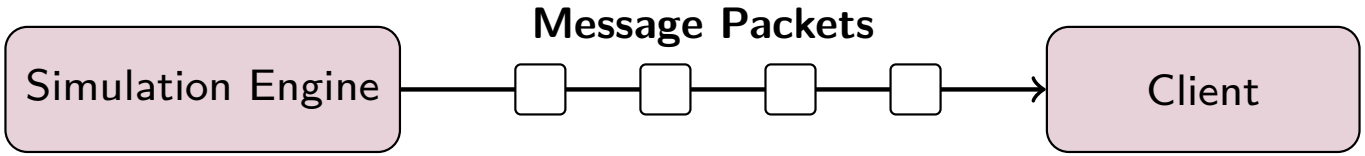
Source Code Modifications

- IMDv3 was implmenetd by appended new methods to existing code modules pertaining to IMDv2
- The IMDv2 code template and strcuture was maintained and only changed where needed
- Enabled the user to switch between v2 and v3 implementations
- The modified source codes were tested for consistency and against unchanged versions of the codebases

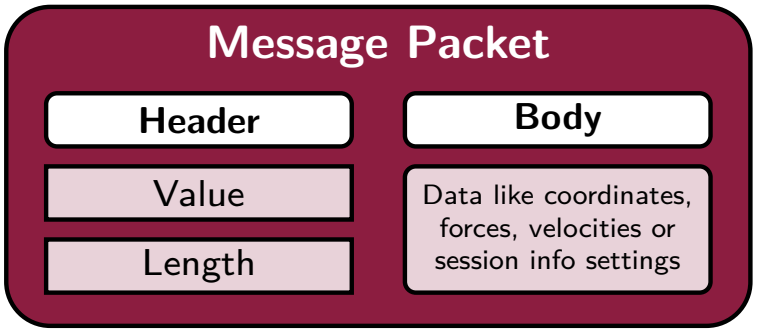
IMDv3 Functionality

- IMDv3 features were added in accordance with the new v3 protocol
- Functions and modules added – provide the ability to stream (send) specific information
- Ability to control streaming settings through MD engine input configuration file

Streaming Data with IMDv3 Protocol



- Data is streamed and sent in the form of packets which contain a *header* and a *body*



- **Value** sets the type of Message/information being sent in the body
- **Length** sets the length of that information
- Simple messages like IMD_GO, IMD_DISCONNECT, IMD_KILL do not have a body
- Others like IMD_FCOORDS or IMD_VELOCITIES have long array-like data within their bodies

IMD Message type	enum	Added in v3?
IMD_DISCONNECT	0	×
IMD_ENERGIES	1	×
IMD_FCOORDS	2	×
IMD_GO	3	×
IMD_HANDSHAKE	**4**	×
IMD_KILL	5	×
IMD_MDCOMM	6	×
IMD_PAUSE	7	×
IMD_TRATE	8	×
IMD_IOERROR	9	×
IMD_SESSIONINFO	**10**	✓
IMD_RESUME	11	✓
IMD_TIME	12	✓
IMD_BOX	13	✓
IMD_VELOCITIES	14	✓
IMD_FORCES	15	✓
IMD_WAIT	16	✓

- **IMD_HANDSHAKE** – special case i.e. length variable contains the version info
- **IMD_SESSIONINFO** – establishes streaming session parameters between MD engine and server

Configuring MD engine input for streaming

- **IMD_SESSIONINFO** – special message packet, contains setting options for streaming process
- Settings configured by the user in simulation input file

```
Header:
  20 (int32) Value: Session info
  7  (int32) Length: Number of
     configuration options in the body

Body:
  <val> (int8) Are line packets sent?
  <val> (int8) Are IMD energy block
    packets sent?
  <val> (int8) Are coords/momenta packets
    sent?
  <val> (int8) Are velocities packets sent
    ?
  <val> (int8) Are force packets sent?
  <val> (int8) Are box packets sent?
  <val> (int8) Are time packets sent?
```

- For eg. NAMD's input file would look like:

```
# IMD parameters
# standard IMD parameters
# IMDon streaming -- on or off
IMDon         yes
# IMDport -- port number to listen on
IMDport       8888
# IMDfreq -- frequency to send data
IMDfreq       1
# IMDwait -- wait for client connection before starting
       simulation
IMDwait       on

# IMD version -- 2 for VMD and 3 for latest protocol
IMDversion    3
# IMD session info settings
# IMDsendPositions -- sending positions of entire system
IMDsendPositions        yes
# IMDsendEnergies -- sending energies and bonded, non-bonded and
        other contributions
IMDsendEnergies         yes
# IMDsendTime -- sending time information (time, dt, step)
IMDsendTime             yes
# IMDsendBoxDimensions -- sending box dimensions (lattice
       vectors a, b, c). If box dimensions are not defined,
       default unit box is sent
IMDsendBoxDimensions    yes
# IMDsendVelocities -- sending velocities of entire system
IMDsendVelocities       yes
# IMDwrapPositions -- wrap positions to box; applicable when
       IMDsendPositions is yes
IMDwrapPositions       no
```

Configuring MD engine input for streaming

- **IMD_SESSIONINFO** – special message packet, contains setting options for streaming process
- Settings configured by the user in simulation input file

```
Header:
  20 (int32) Value: Session info
  7  (int32) Length: Number of
     configuration options in the body

Body:
  <val> (int8) Are line packets sent?
  <val> (int8) Are IMD energy block
    packets sent?
  <val> (int8) Are coords/momenta packets
    sent?
  <val> (int8) Are velocities packets sent
    ?
  <val> (int8) Are force packets sent?
  <val> (int8) Are box packets sent?
  <val> (int8) Are time packets sent?
```

- For eg. NAMD's input file would look like:

NAMD Scalable Molecular Dynamics

```
# IMD parameters
# standard IMD parameters
# IMDon streaming -- on or off
IMDon         yes
# IMDport -- port number to listen on
IMDport       8888
# IMDfreq -- frequency to send data
IMDfreq       1
# IMDwait -- wait for client connection before starting
       simulation
IMDwait       on

# IMD version -- 2 for VMD and 3 for latest protocol
IMDversion    3
# IMD session info settings
# IMDsendPositions -- sending positions of entire system
IMDsendPositions        yes
# IMDsendEnergies -- sending energies and bonded, non-bonded and
       other contributions
IMDsendEnergies         yes
# IMDsendTime -- sending time information (time, dt, step)
IMDsendTime             yes
# IMDsendBoxDimensions -- sending box dimensions (lattice
       vectors a, b, c). If box dimensions are not defined,
       default unit box is sent
IMDsendBoxDimensions    yes
# IMDsendVelocities -- sending velocities of entire system
IMDsendVelocities       yes
# IMDwrapPositions -- wrap positions to box; applicable when
       IMDsendPositions is yes
IMDwrapPositions       no
```

Configuring MD engine input for streaming

- **IMD_SESSIONINFO** – special message packet, contains setting options for streaming process
- Settings configured by the user in simulation input file

```
Header:
  20 (int32) Value: Session info
  7  (int32) Length: Number of
     configuration options in the body

Body:
  <val> (int8) Are line packets sent?
  <val> (int8) Are IMD energy block
    packets sent?
  <val> (int8) Are coords/momenta packets
    sent?
  <val> (int8) Are velocities packets sent
    ?
  <val> (int8) Are force packets sent?
  <val> (int8) Are box packets sent?
  <val> (int8) Are time packets sent?
```

- Similar input functionality available in other MD engines

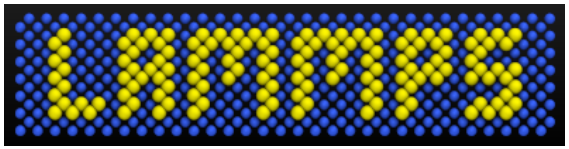
- For eg. NAMD's input file would look like:

NAMD Scalable Molecular Dynamics

```
# IMD parameters
# standard IMD parameters
# IMDon streaming -- on or off
IMDon         yes
# IMDport -- port number to listen on
IMDport       8888
# IMDfreq -- frequency to send data
IMDfreq       1
# IMDwait -- wait for client connection before starting
       simulation
IMDwait       on

# IMD version -- 2 for VMD and 3 for latest protocol
IMDversion    3
# IMD session info settings
# IMDsendPositions -- sending positions of entire system
IMDsendPositions        yes
# IMDsendEnergies -- sending energies and bonded, non-bonded and
        other contributions
IMDsendEnergies         yes
# IMDsendTime -- sending time information (time, dt, step)
IMDsendTime             yes
# IMDsendBoxDimensions -- sending box dimensions (lattice
       vectors a, b, c). If box dimensions are not defined,
       default unit box is sent
IMDsendBoxDimensions    yes
# IMDsendVelocities -- sending velocities of entire system
IMDsendVelocities       yes
# IMDwrapPositions -- wrap positions to box; applicable when
       IMDsendPositions is yes
IMDwrapPositions        no
```

Configuring MD engine input for streaming



- The input file in LAMMPS can be set with a line that takes care of various IMDv3 settings

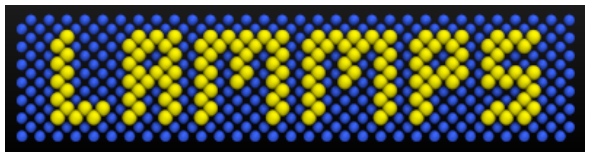
```
## IMD settings
# https://docs.lammps.org/fix_imd.html
fix 2 all imd 8888 trate 1 version 3 unwrap off time on box on coordinates on velocities on forces on
```

FAST. FLEXIBLE. FREE.
GROMACS

- For GROMACS the mdp file is modified with requisite IMDv3 settings, which can then be assembled into a binary tpr using grompp

```
; IMD parameters
IMD-group    = System    ; group to send to IMD
IMD-nst      = 1         ; transmission rate to IMD
IMD-version  = 3         ; version of IMD protocol
IMD-time     = yes       ; if time information should be
      sent (time, dt, step)
IMD-box      = yes       ; if box dimensions should be sent
      (lattice vectors a, b, c)
IMD-coords   = yes       ; if coordinates should be sent
IMD-unwrap   = no        ; if coordinates should be
      unwrapped
IMD-vels     = yes       ; if velocities should be sent
IMD-forces   = yes       ; if forces should be sent
IMD-energies = yes       ; if energies should be sent
```

Availability of Source Codes and Docker containers



- **lammps/lammps**
  `github.com/lammps/lammps`
- *Part of official release*

NAMD
Scalable Molecular Dynamics

- **tcbgUIUC/namd**
  `gitlab.com/tcbgUIUC/namd`
- *Part of official source code repository, will be part of NAMD release 3.1*

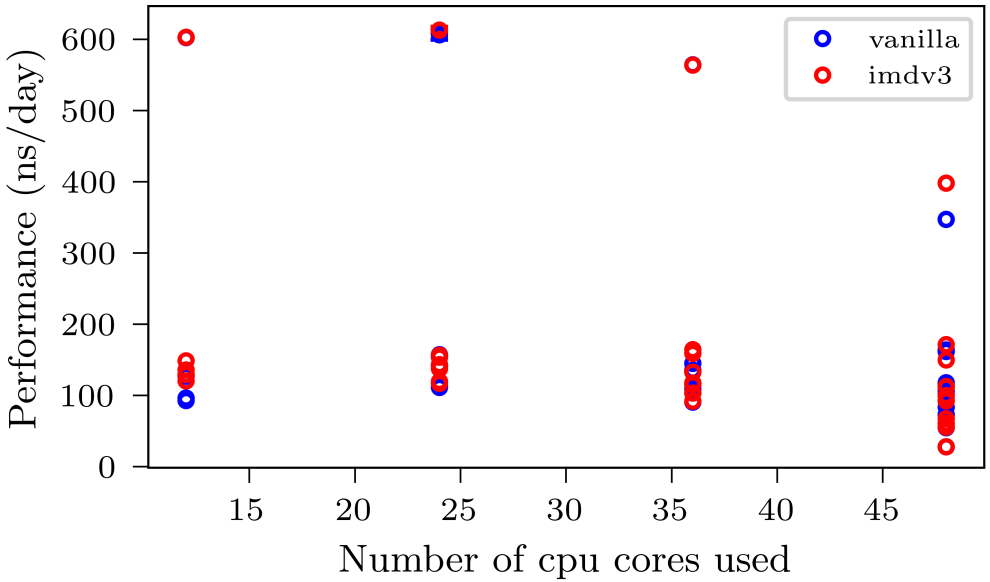
FAST. FLEXIBLE. FREE.
GROMACS

- **heydenlabasu/streaming-md/gromacs**
  `gitlab.com/heydenlabasu/streaming-md/gromacs/-/tags/imdv3-sans-tests`
- In process of getting merged it into official repository

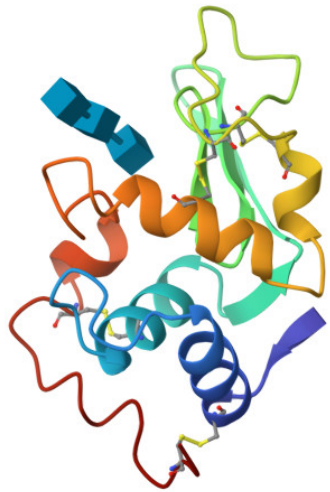
- **Becksteinlab/streaming-md-docker**
  `github.com/Becksteinlab/streaming-md-docker`

Does IMDv3 affect MD engine performance?

- We perform benchmarking analysis for IMDv3-modified version of our code with streaming
- Choose test system: HEWL in water (30,000 atom system)
- We limit our tests to certain computational resources – 1 exclusive node on ASU's SOL supercomputer - 1 A100 GPU and 48 CPU cores.
- First, test for optimized run setting for this system and compare with `vanilla` code.
- As an example, we show detailed results for GROMACS



GROMACS optimization results

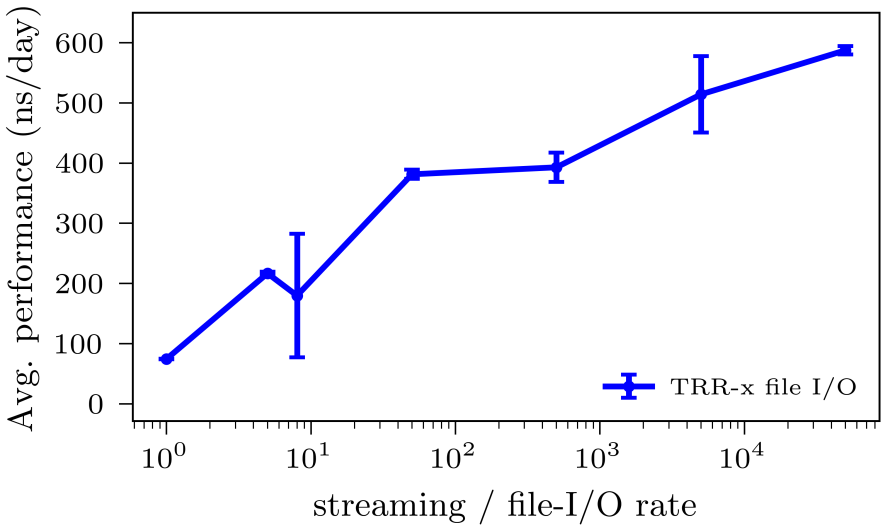


Hen Egg-White Lysozyme (HEWL)

How good is IMDv3 streaming?

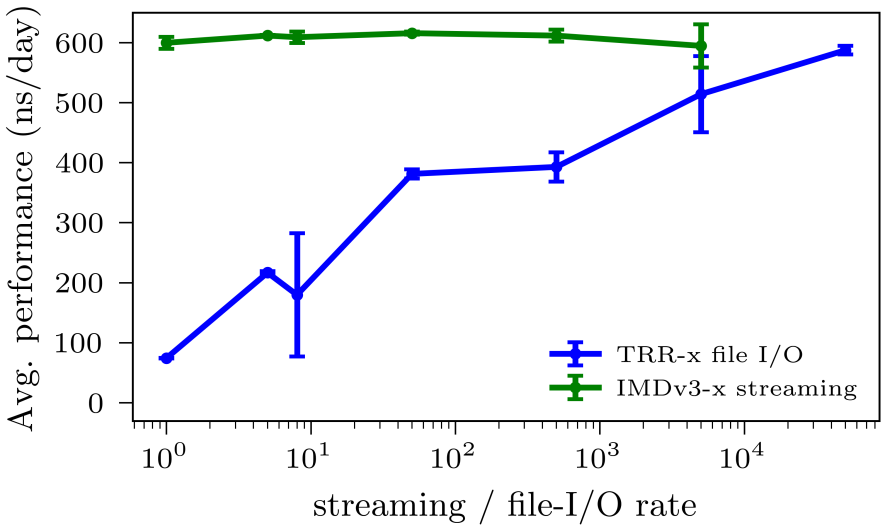
- Traditional alternative to streaming is writing/reading data to/from files i.e file I/O
- We compare performance of IMDv3 streaming vs file I/O

How good is IMDv3 streaming?



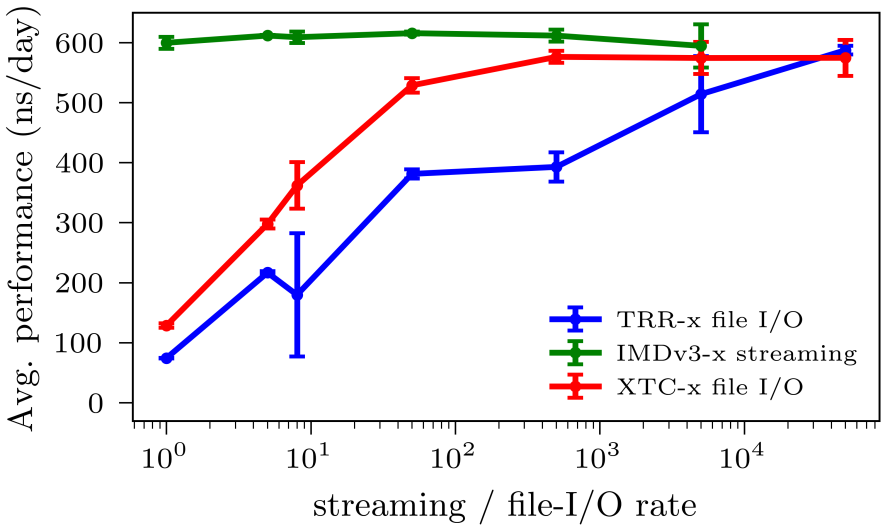
- Traditional alternative to streaming is writing/reading data to/from files i.e file I/O
- We compare performance of IMDv3 streaming vs file I/O

How good is IMDv3 streaming?



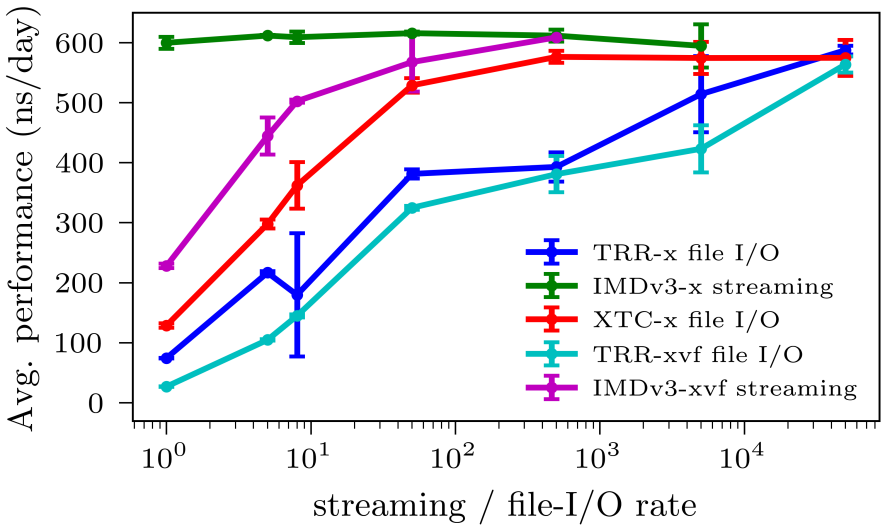
- Traditional alternative to streaming is writing/reading data to/from files i.e file I/O
- We compare performance of IMDv3 streaming vs file I/O
- Streaming significantly outperforms when `positions` are being streamed/written

How good is IMDv3 streaming?



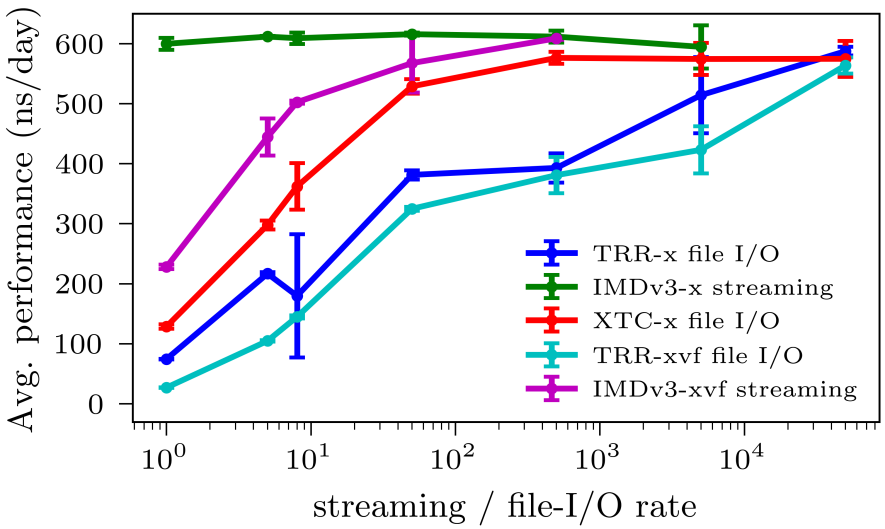
- Traditional alternative to streaming is writing/reading data to/from files i.e file I/O
- We compare performance of IMDv3 streaming vs file I/O
- Streaming significantly outperforms when `positions` are being streamed/written
- Writing to binary formats (`.xtc`) improves file I/O speeds

How good is IMDv3 streaming?



- Traditional alternative to streaming is writing/reading data to/from files i.e file I/O
- We compare performance of IMDv3 streaming vs file I/O
- Streaming significantly outperforms when `positions` are being streamed/written
- Writing to binary formats (`.xtc`) improves file I/O speeds
- When multiple data types are being streamed/written, streaming still outperforms file I/O

How good is IMDv3 streaming?



- Traditional alternative to streaming is writing/reading data to/from files i.e file I/O
- We compare performance of IMDv3 streaming vs file I/O
- Streaming significantly outperforms when `positions` are being streamed/written
- Writing to binary formats (`.xtc`) improves file I/O speeds
- When multiple data types are being streamed/written, streaming still outperforms file I/O
- Similar qualitative results when testing LAMMPS and NAMD

Conclusion

- We have successfully implemented and tested IMDv3 in 3 MD engines (GROMACS, LAMMPS, NAMD)
- The protocol allows enables various features, most importantly user-controlled flexible data streaming of various simulation properties
- Source code and Docker containers are available for easy adoption
- Tests show that IMDv3-modifedi MD engine code performs as well as the unmodified versions
- Benchmarks show IMDv3 streaming is efficient and outperforms file I/O
  - For certain scenarios streaming had negligible affect on performance of a running simulation
- IMDv3 can be used for many practical, high-performance molecular simulation based workflows

Questions?

Acknowledgements

Lab Members at ASU

- Lawson Woods
- Heekun Cho
- Matthias Heyden
- Oliver Beckstein

Collaborators at MDAnalysis

- Hugo MacDermott-Opeskin
- Jennifer Clark
- Yuxuan Zhuang
- Irfan Alibay

Thank you!

Happy to take any questions :)