MVCC Software Transactional Memory in Rust

Jiakun Fan

University of Rochester

Content

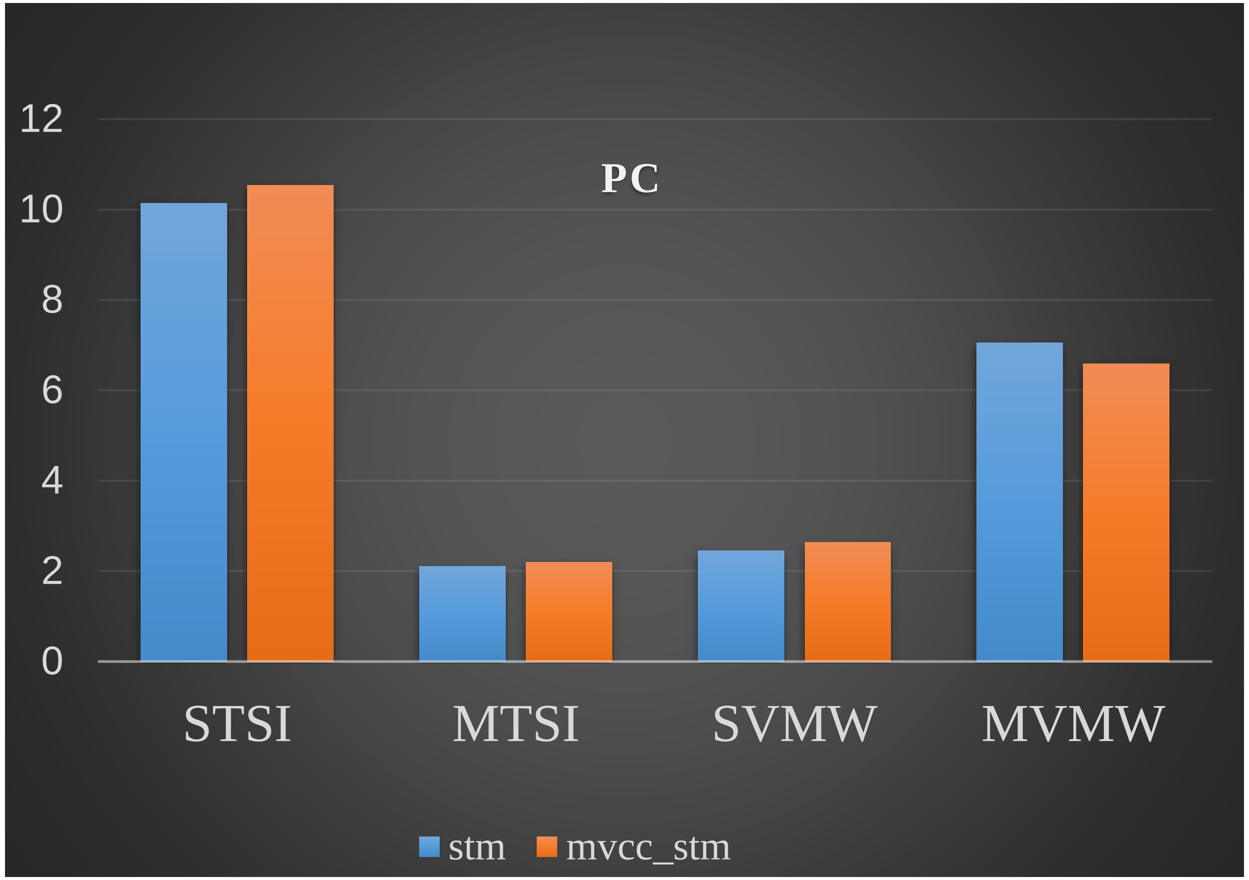
- Evaluation
- Discussion
- Conclusion
- Future Work

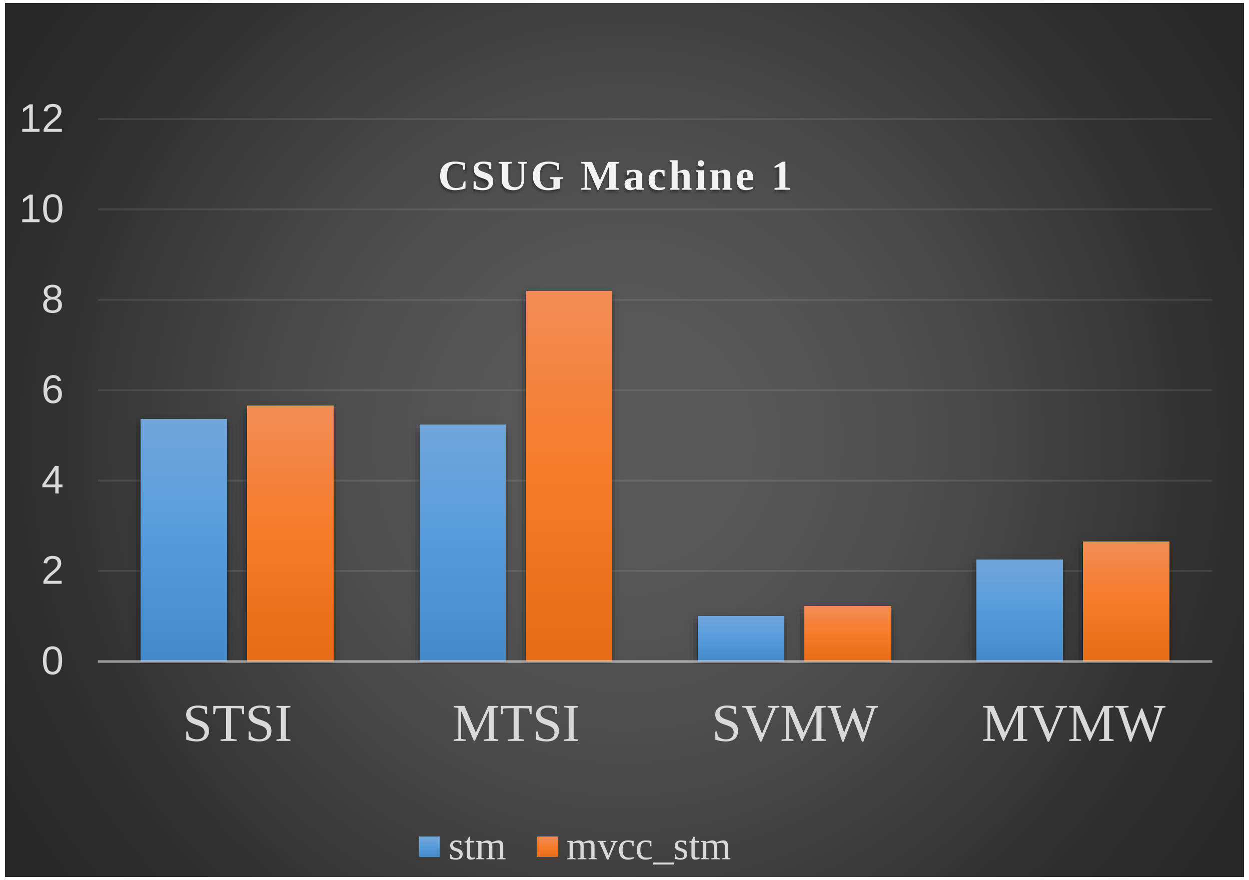
Evaluation

Test Cases

- STSI (single thread self increment) e.g. a = a + 1
- MTSI (multiple threads self increment)
- SVMW (single variable multiple writers)
- MVMW (multiple variables multiple writers)

Result


PC
12
10
8
6
4
2
0
STSI
MTSI
SVMW
MVMW
stm
mvcc_stm


CSUG Machine 1
12
10
8
6
4
2
0
STSI
MTSI
SVMW
MVMW
stm
mvcc_stm

Discussion

- In most cases, stm is faster than mvcc stm (though only a little faster)
- Speedup is too small to cover overhead.
- While mvcc stm may experience a slight speedup in rare cases, this improvement is dependent on the specific machine being used.
- Due to the variability in write order across multiple runs, it is challenging to accurately assess the speed of the process.
- Retry early is not a good optimization.

Conclusion

- MVCC is not a general method to speedup stm.
- It provides concurrent access to the STM

Future Work

✓ MVCC STM

✓ Support operations with side-effects (esp. print)

□ Implementing the blocking facility

□ Find a general speedup solution (if have)

Thank you!

github url: https://github.com/chosen-ox/vincent_stm