Automatic Reflex-Based Balancing for Small Humanoid Robots

A Thesis Presentation
by

Sara McGrath

Overview

- Motivation
- Background
- Literature and Correction Algorithms
- Tuning the Corrections
- Evaluation
- Conclusions and Future Work
- Questions

Motivation

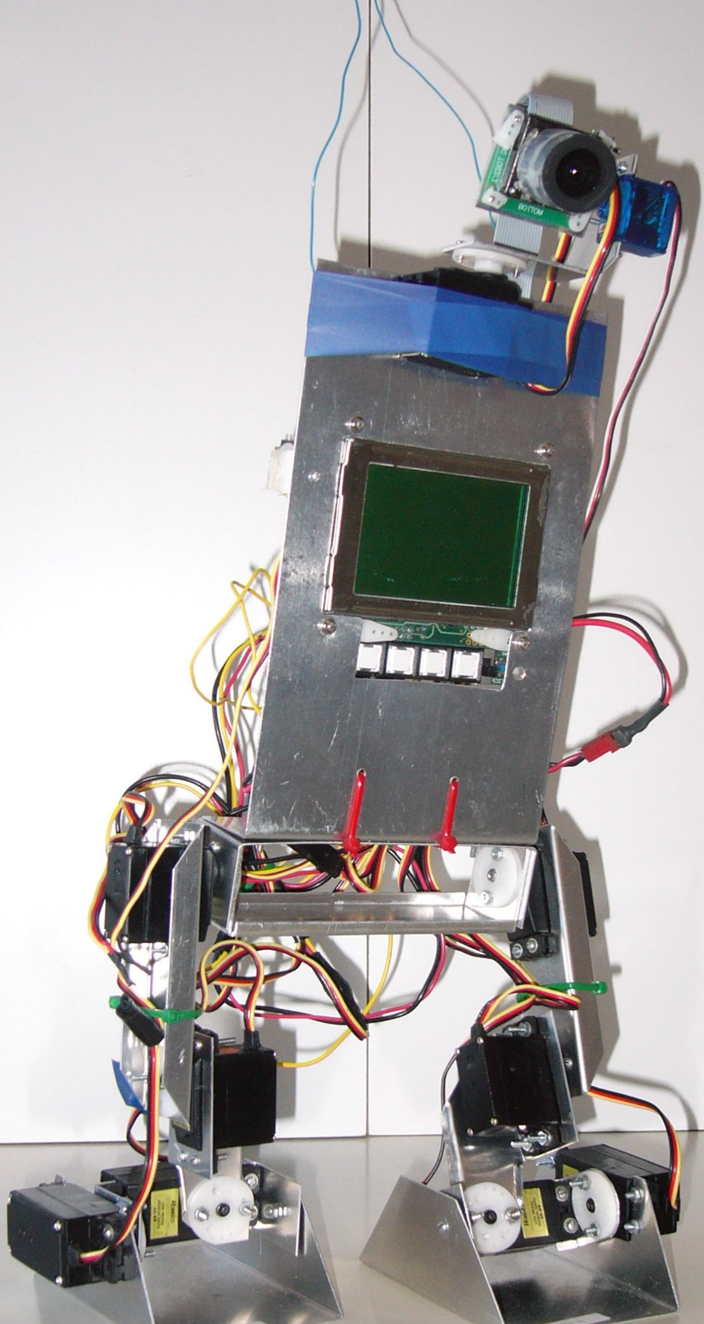
- Humanoid robots are general solutions
  - but need to move over many terrains
- Need to add balancing, but how?
  - Similar to humans: use reflexes with feedback
  - subconscious balancing
- Use single sensor, basic robot for simple, robust algorithm

Research Questions

1. Can a balancing reflex that is a tightly coupled feedback loop implemented on a single motion sensor dynamically balance a small humanoid robot in real-time? If so, can the robot measurably improve its gait with this?

Research Questions

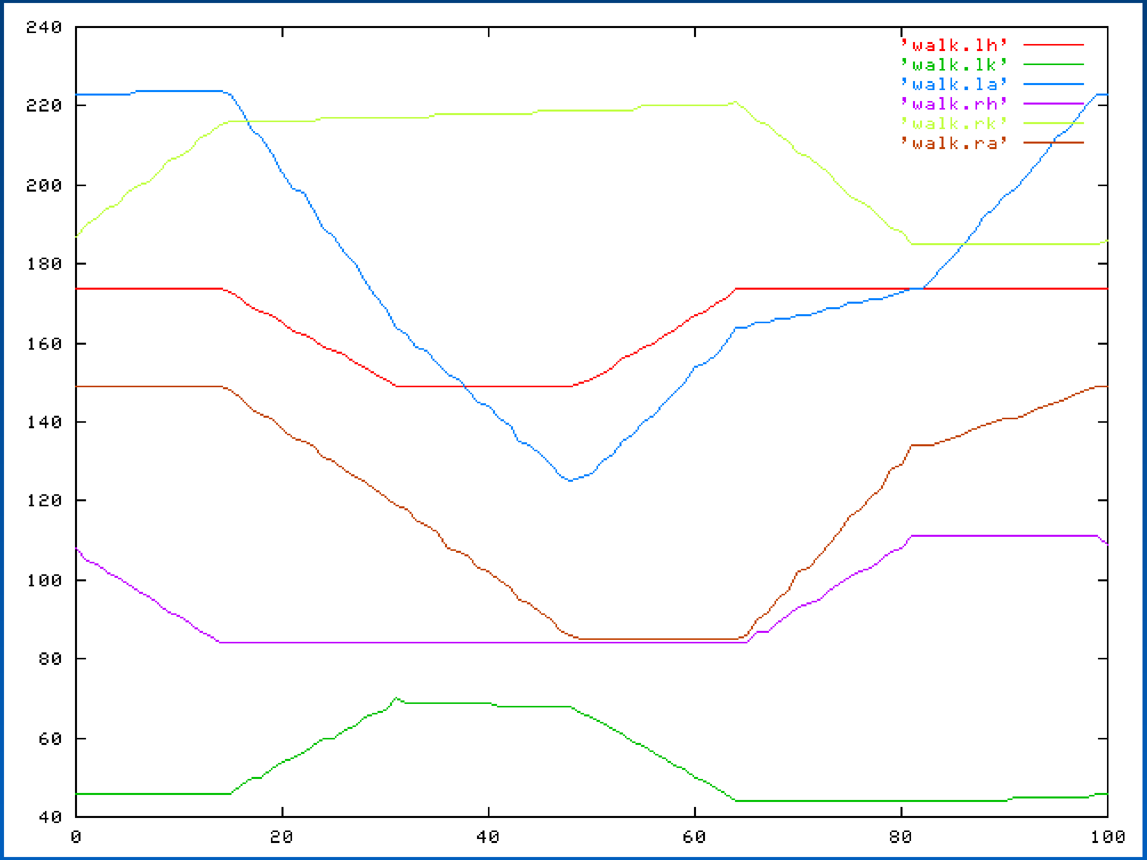
2. What balancing reflex should be used to dynamically balance a robot? What are the strengths and weaknesses of the various algorithms? How does the algorithm previously developed for the gyroscope compare to a standard control algorithm?



Lillian

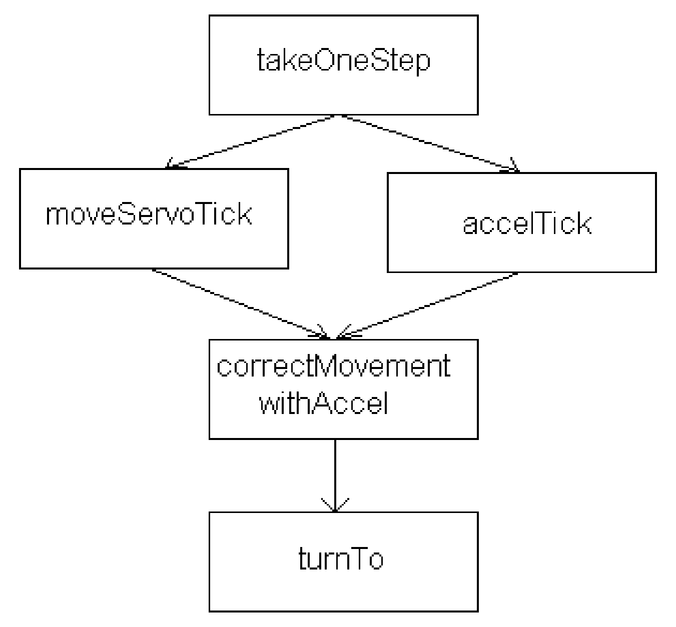
- 8 DOF
- Eyebot Controller
- Accelerometer

Gait Creation



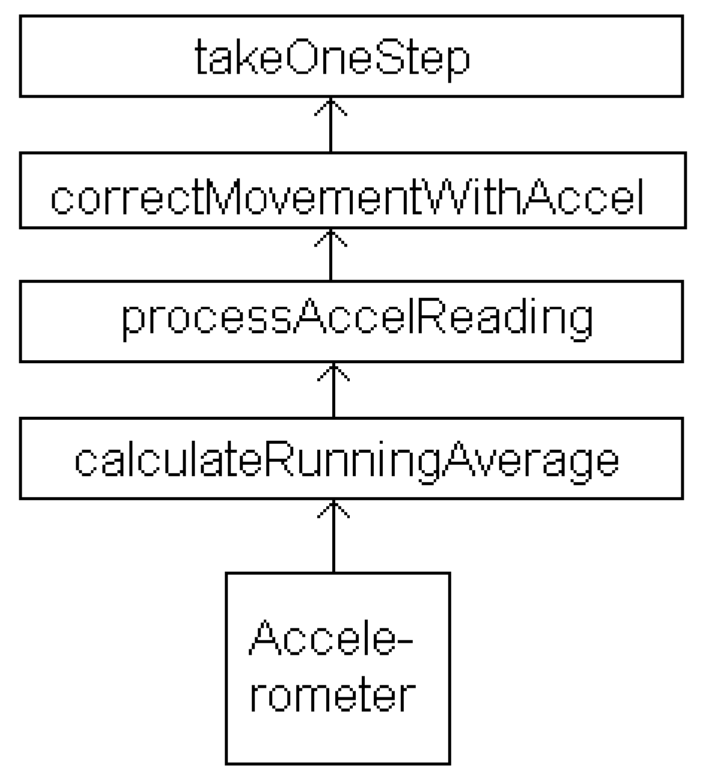
- Each line is a joint's settings
- Interpolation used between saved points
- Adjustments applied to base position

Gait Creation



- Corrections applied to interpolated position
- Setting sent to joint to be applied
- Each 'tick' is one OS count, about 1 ms

Corrections



- Accelerometer read constantly
- Readings averaged, adjusted for zero point
- Corrections made based on percieved error

Literature

- Normally use special purpose algorithms
  - COM
  - ZMP
- Beijing University used reflexes first
  - ZMP, landing and posture reflex
  - sensors trigger correction to pre-calculated walk

Literature

- KMUTT used velocity controllers
  - PD controller used to balance as needed; velocity at hip used for dynamic walk
  - PD controller for hip height used with force sensors for static walk
- Tao-Pie-Pie used gyroscopes only
  - threshold balancing adjusted walk
  - always on

Correction Methods

- Three main methods: PID (P), Threshold (T), and Hybrid (H)
- All correction methods needed to control the rate of correction to prevent oscillation
- All methods also used a preset accelerometer reading baseline to calculate error

Correction Methods: P

- P: Proportional Integral Derivative Controller
- standard control mechanism
- corrections mainly based on a Proportion of the error, but also a percentage of the Integral and Derivative errors
- baseline is a single setpoint for any given time point
- quick reaction; more likely to overcorrect

Correction Methods: T

- T based on threshold boundaries
- baseline is a pair of thresholds used to specify the desired accelerometer reading range at a given time
- minimal correction made when error crosses a threshold by more than a preset amount
- slower reaction, but less likely to overcorrect

Correction Methods: H

- H is a hybrid combination of the two
- baseline is a single setpoint for any given time point
- error correction is T until a preset error amount (double the normal boundary) and P thereafter
- theory is to combine the best features of the two prior methods
- must be tuned after the other two methods

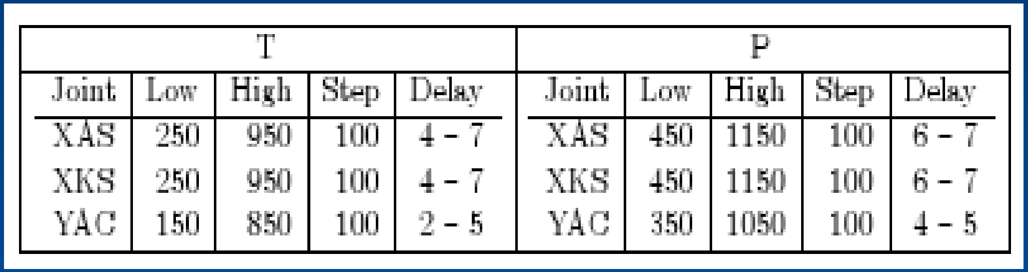
SAE

- SAE (Sum of Absolute Error) used to quantify test results
- Absolute error from baseline summed to measure the goodness of a walk
- SSE (Sum of Squared Error) not used as it gives too much weight to outlier data points, overwhelming information from lesser errors

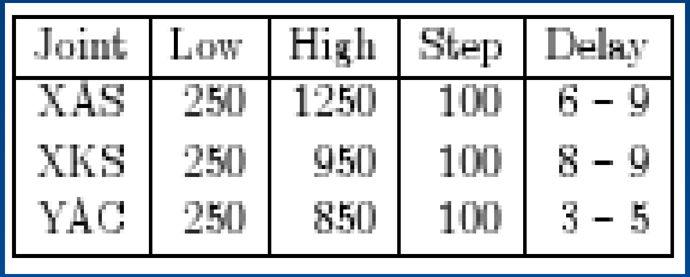
Tuning: Walk

- Lillian tilted on platform for 30° in each direction, angular velocity of 240°
- tilting used to determine initial tuning ranges for P, T
- single joints tested first, then multiple

Tuning: Walk



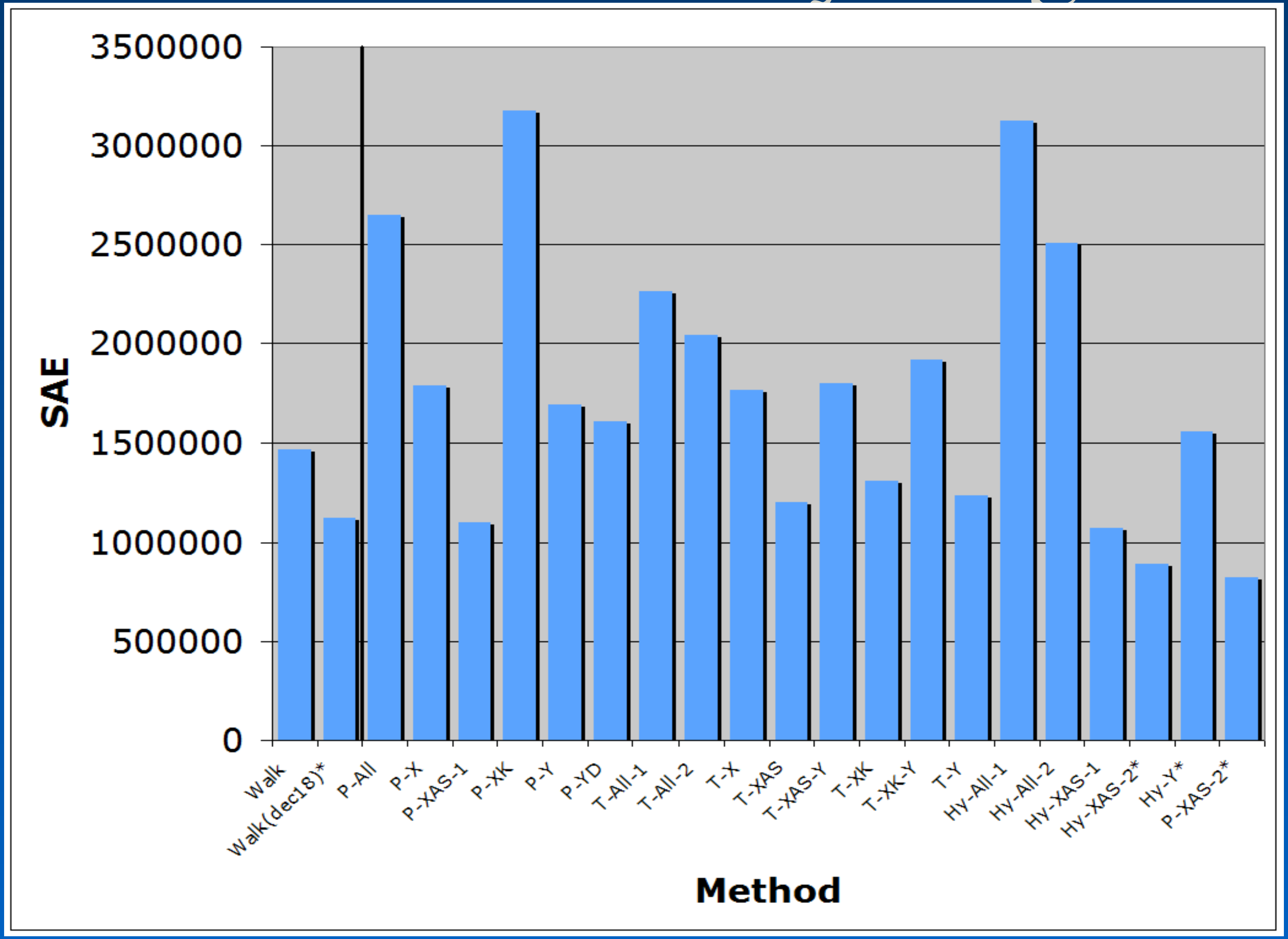
T					P				
Joint	Low	High	Step	Delay	Joint	Low	High	Step	Delay
XAS	250	950	100	4 - 7	XAS	450	1150	100	6 - 7
XKS	250	950	100	4 - 7	XKS	450	1150	100	6 - 7
YAC	150	850	100	2 - 5	YAC	350	1050	100	4 - 5



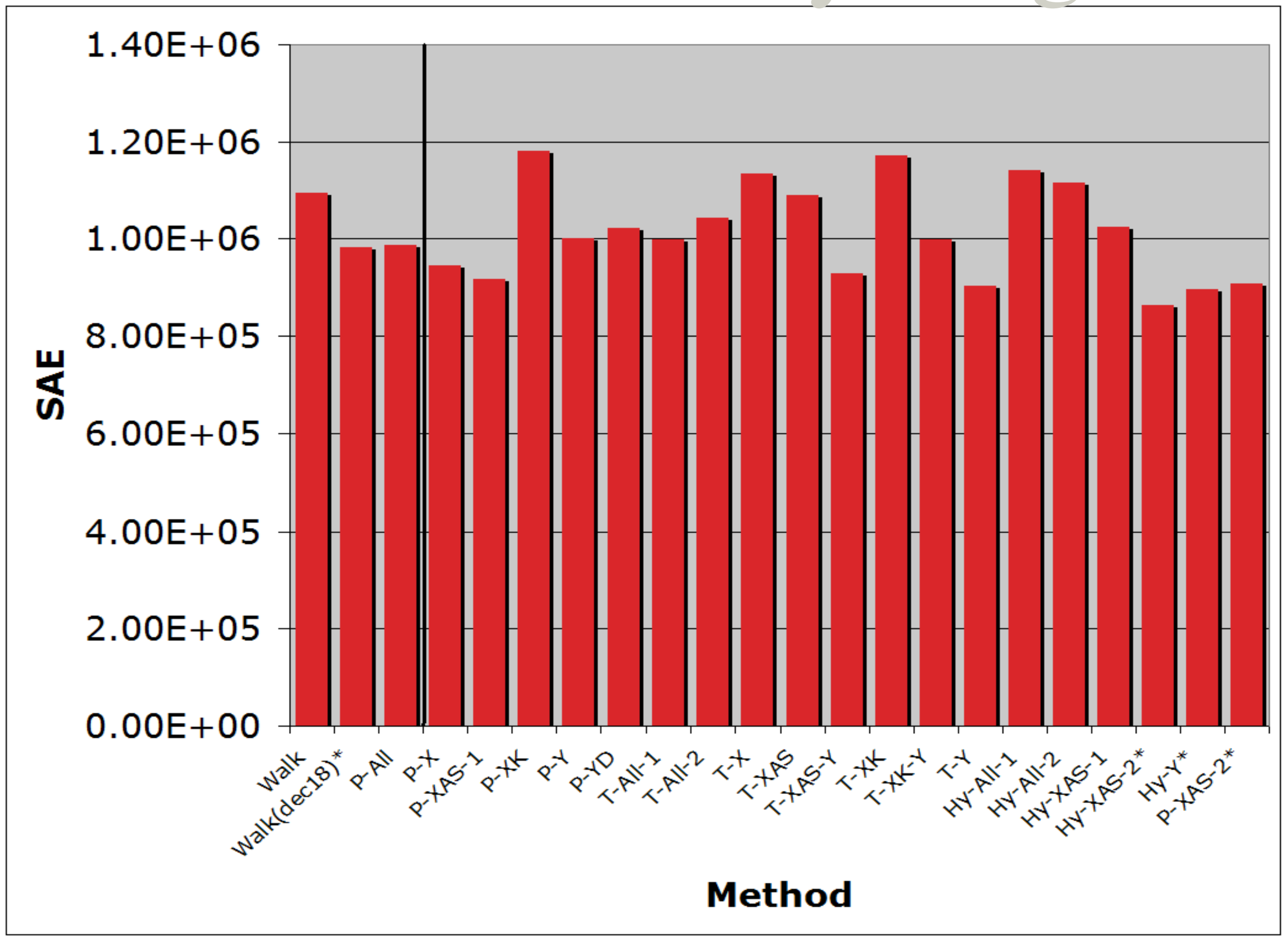
Joint	Low	High	Step	Delay
XAS	250	1250	100	6 – 9
XKS	250	950	100	8 – 9
YAC	250	850	100	3 – 5

- Walk used after stand testing; finer grained exploration of test space
  - Single joints used again to start
- not all tests able to be completed: higher movement speed of walk made some settings untenable

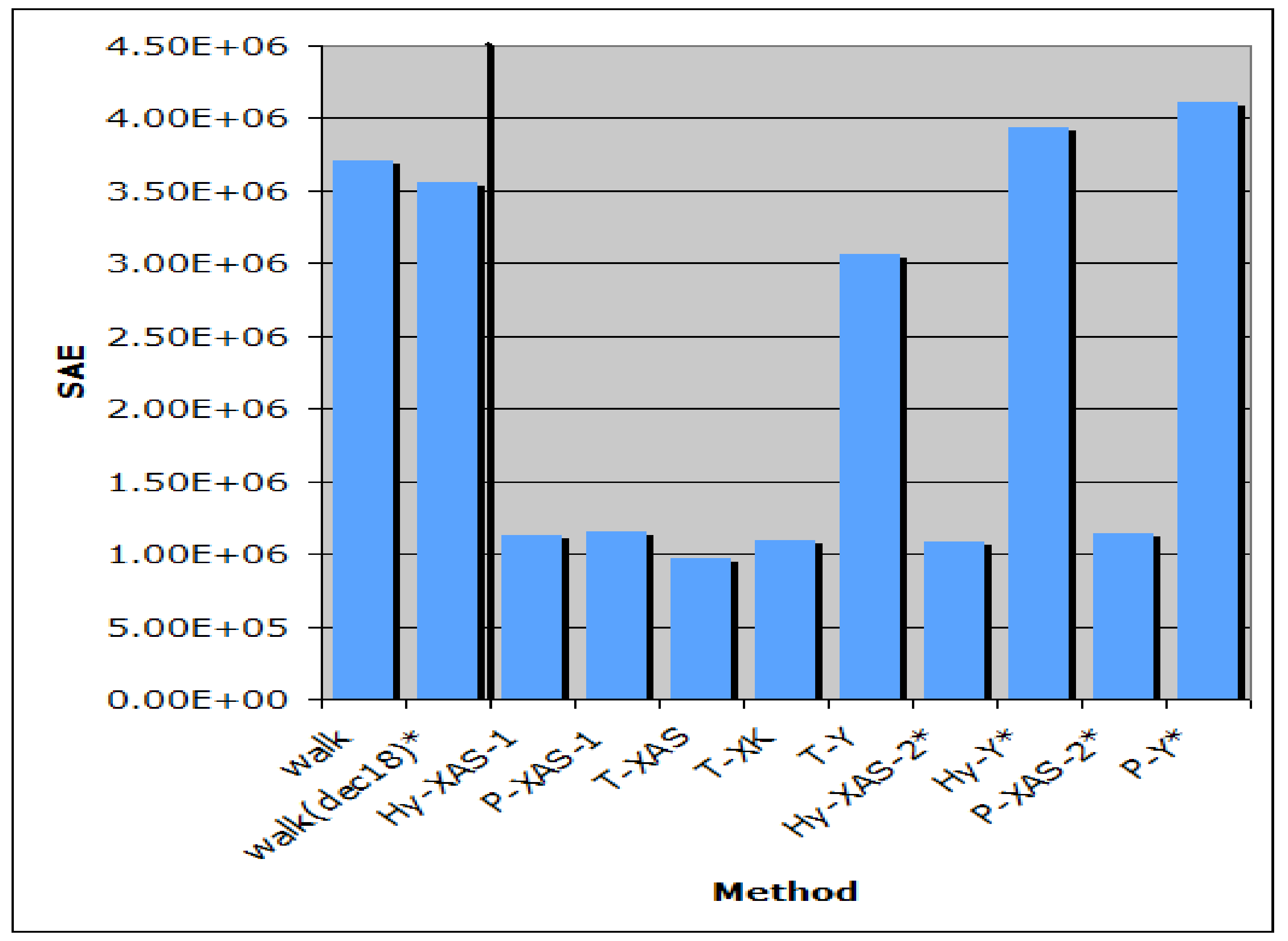
Evaluation: Everything X



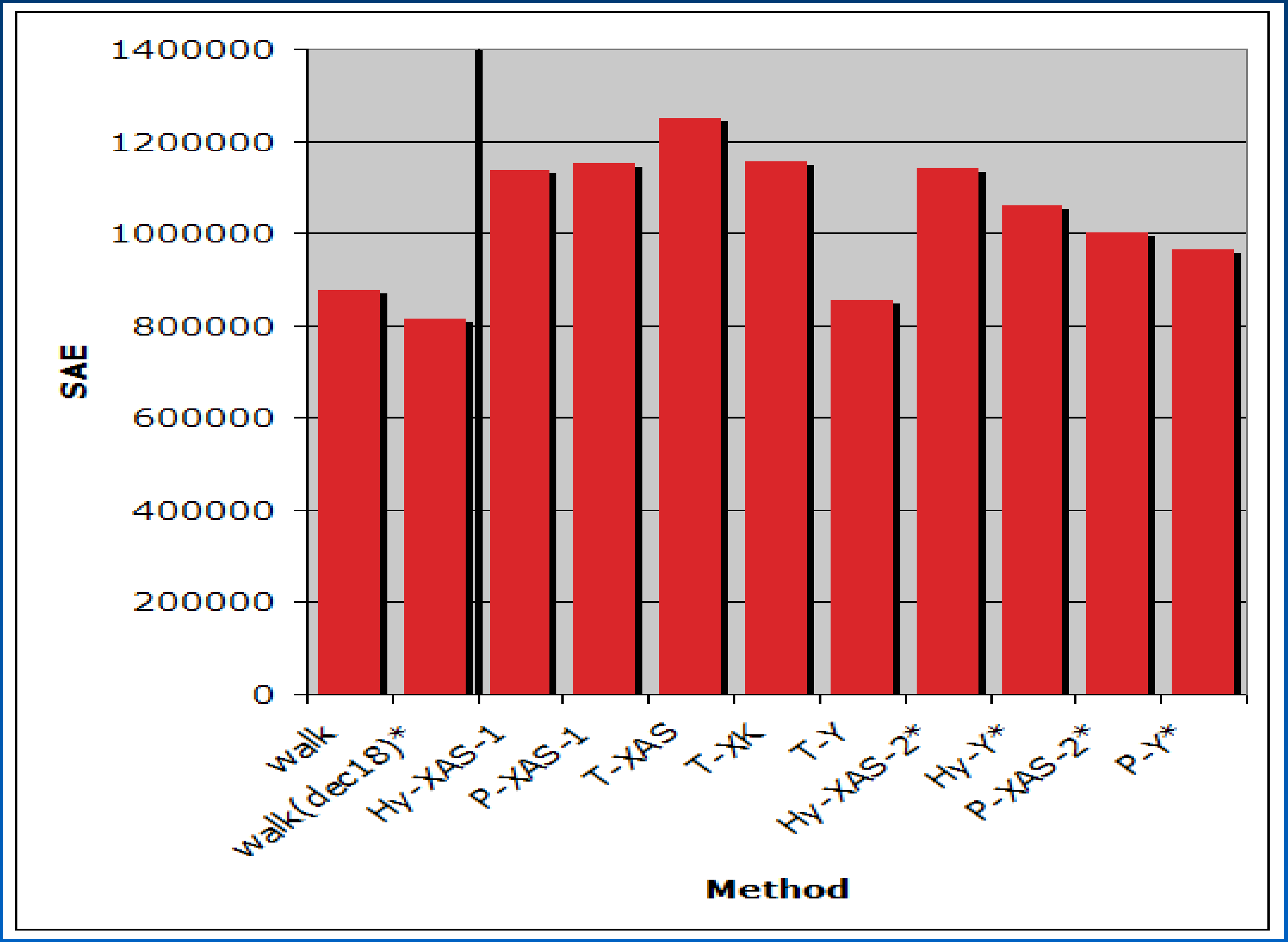
Evaluation: Everything Y



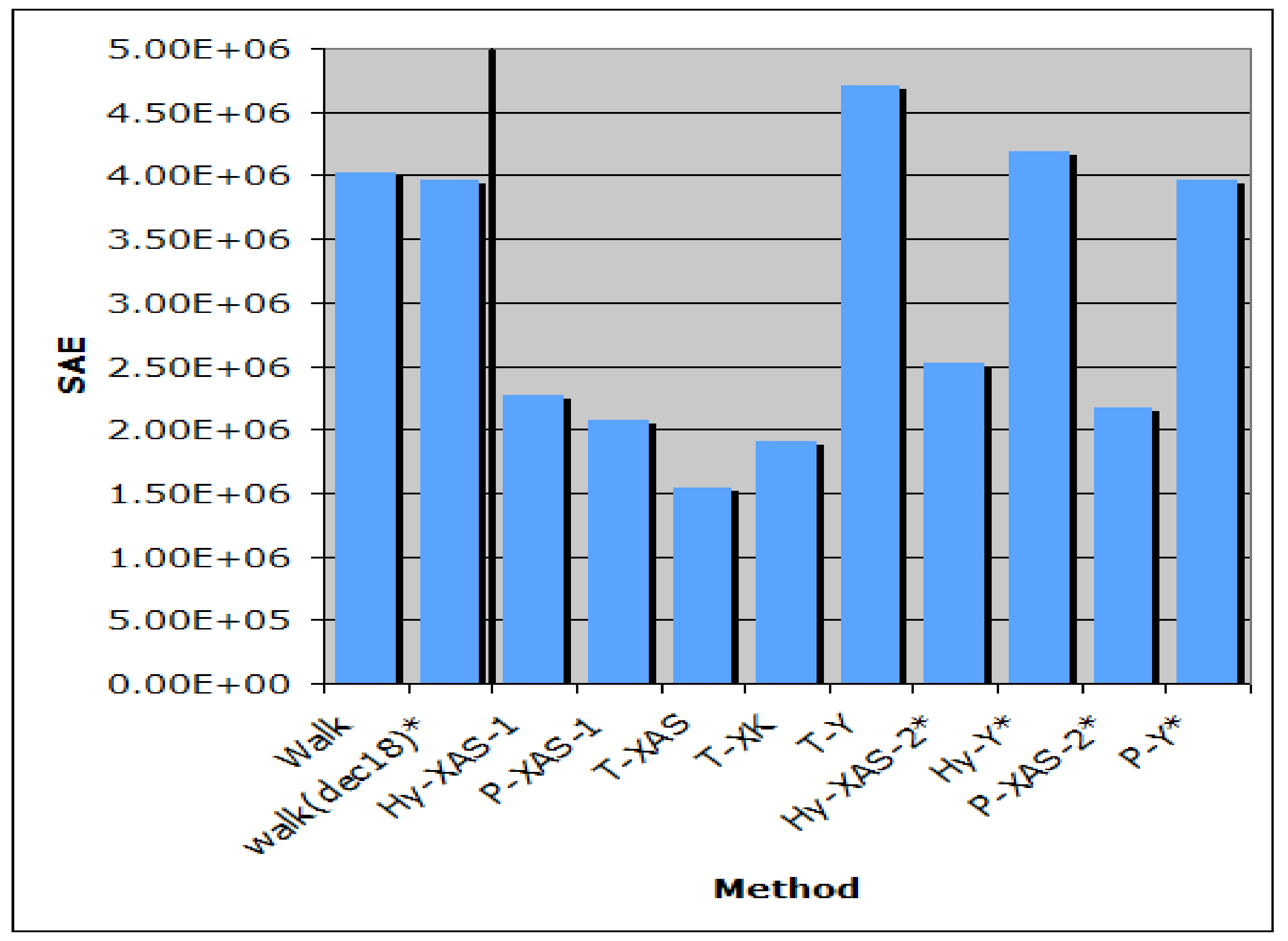
Evaluation: Random Walk 5 X



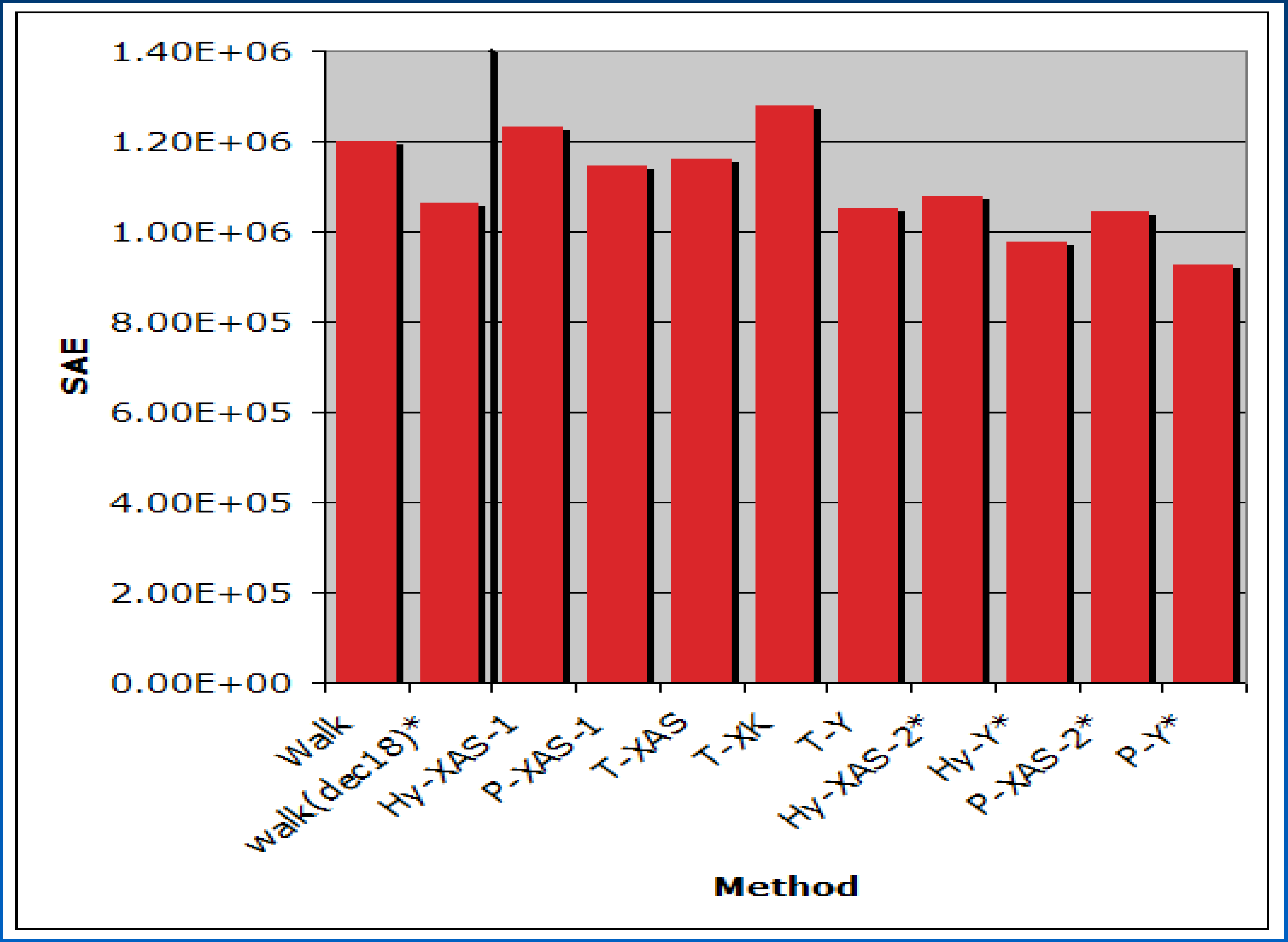
Evaluation: Random Walk 5 Y



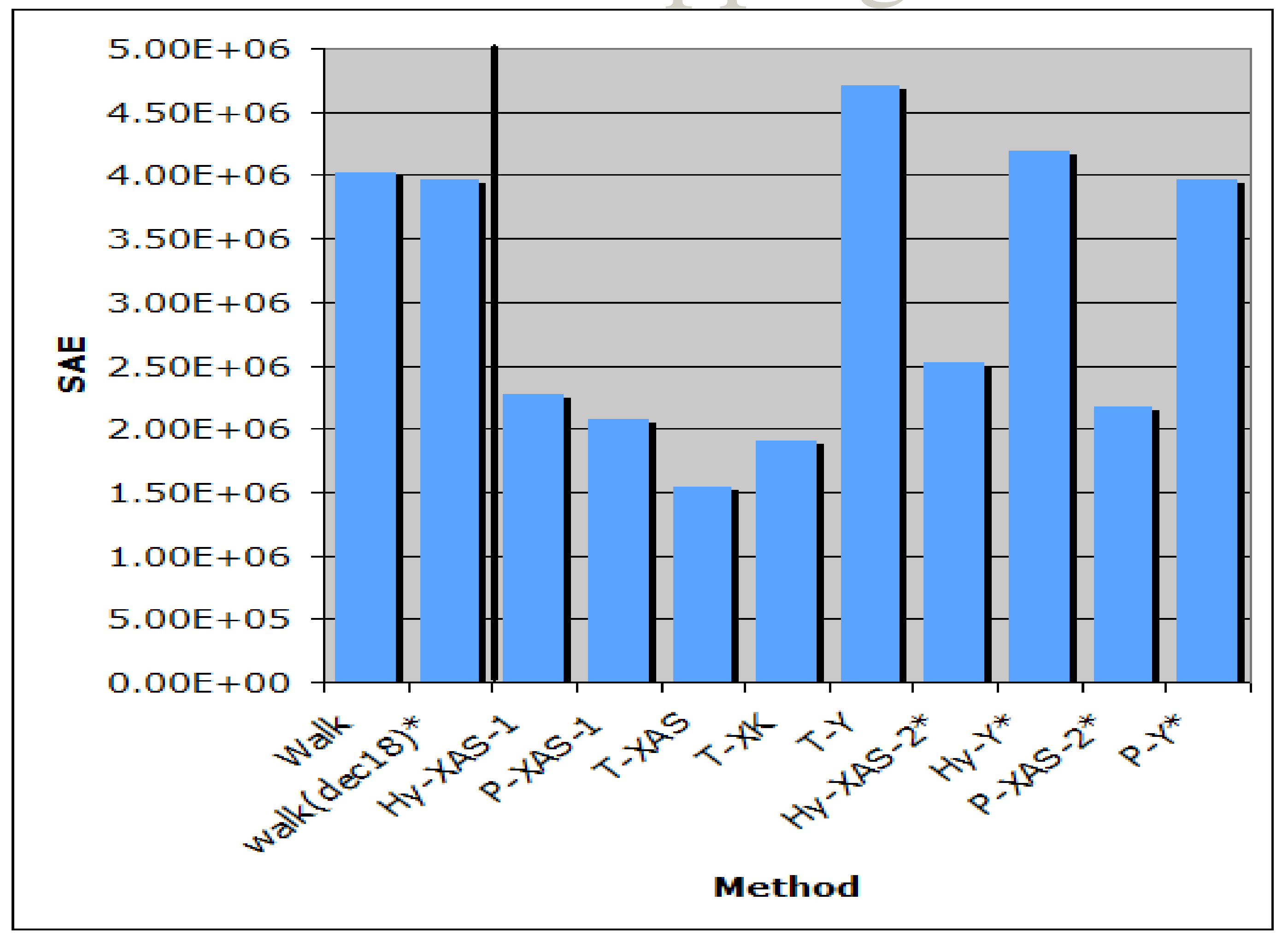
Evaluation: Random Walk 10 X



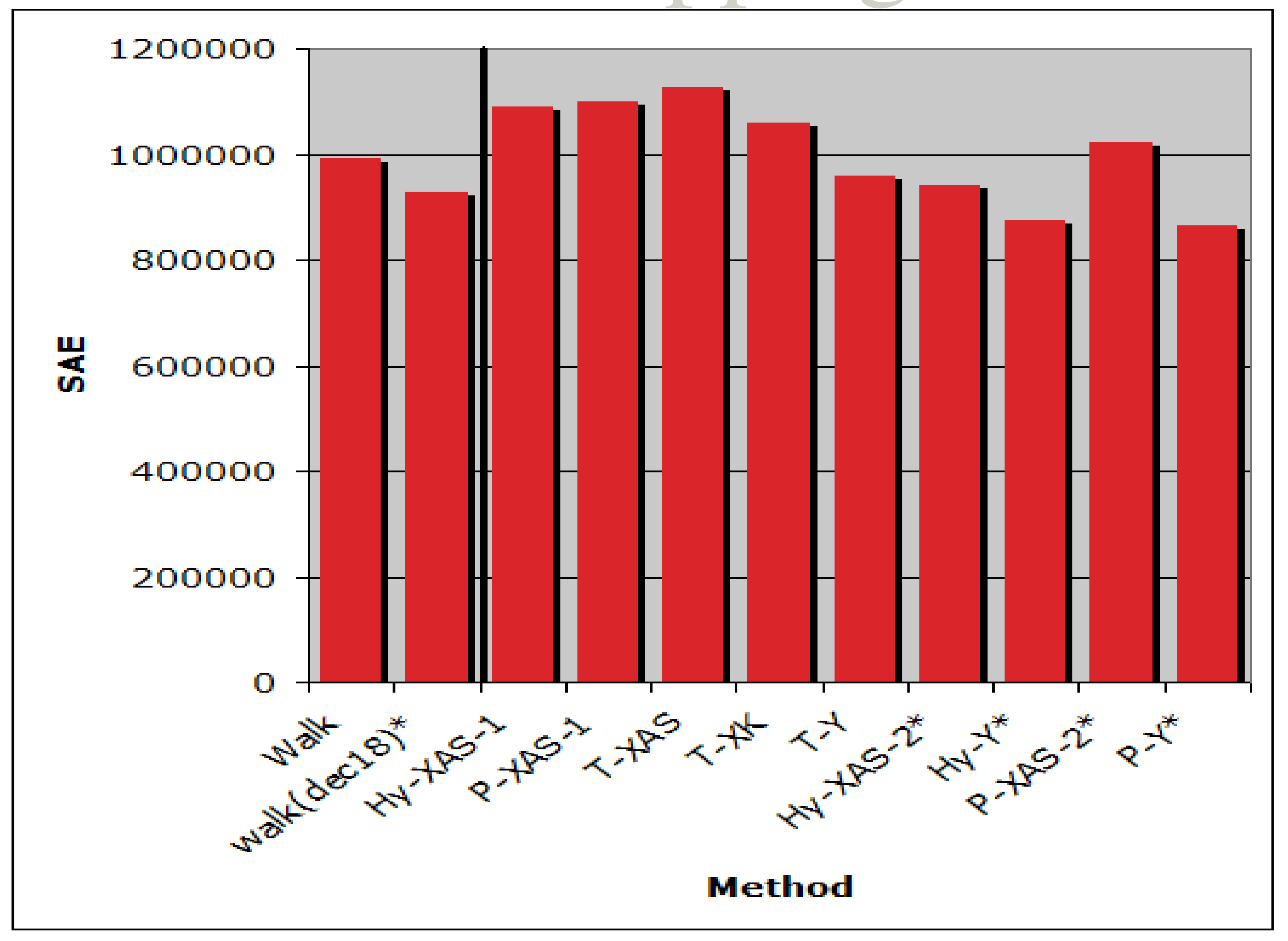
Evaluation: Random Walk 10 Y



Evaluation: Stepping Field X



Evaluation: Stepping Field Y



Conclusions

- Balancing reflexes with simple algorithms provide enough information to balance
- A single sensor tied to a reflex is sufficient
- Balancing has limits: like with humans; best on tilting surfaces, not as good on uneven terrain

Conclusions

- Different algorithms have different strengths
  - P improves with more complex terrain
  - T best on RW, tilts
  - H just generally ok, never best, not usually worst
- Any algorithm is better than none, except on the stepping field
- T is easiest to use, relatively similar results

Future Work

- Remove assumption that joint movement is correlated
- Adjust threshold boundaries
- Physically modify robot
  - ie, add camera
- Transfer tests to a different robot

Acknowledgements

- My supervisor, Dr. Jacky Baltes
- Dr. Anderson and Dr. Kriellaars, my thesis committee
- Autonomous Agents Lab
- Department of Computer Science
- Dominic Leung and Kevin Lambrecht
- Friends and family

Questions?