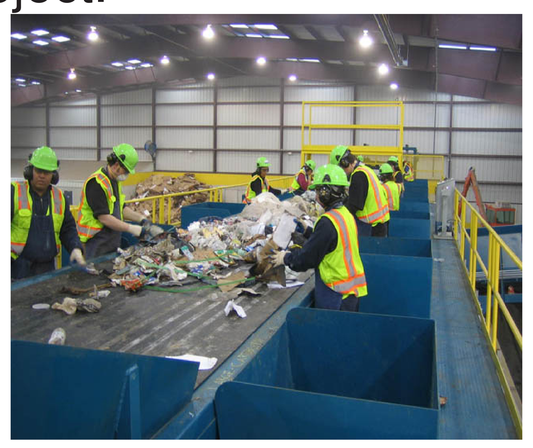
Minimum Time Control in SCARA Robot Simulation

Sam Tormey, Nick Link, Rick LeVan. Advisor: Matthias Heinkenschloss

Department of Computational and Applied Mathematics, Rice University

Introduction

Improving the efficiency of recycling through automation was the inspiration for our project.



We model the removal of objects from a conveyor belt with a MATLAB simulation of a SCARA robot.
We solve a constrained optimization problem to find optimal (minimum-time) path between starting and ending robot configurations.

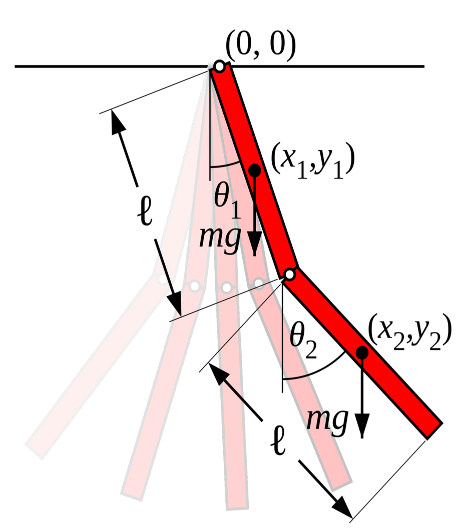
Double Pendulum Physics

We calculate the kinetic energy of our system as

$$E = \frac{1}{2}\dot{\theta}_1^2(I_4 + 2I_5\cos\theta_2) + \frac{1}{2}\dot{\theta}_2^2 I_6 + \dot{\theta}_1\dot{\theta}_2(I_3 + I_5\cos\theta_2)$$

where the I_j terms represent moments of inertia.
We ignore gravity and thus potential energy. The Euler-Lagrange equation then applies only to the kinetic energy above:

$$\frac{\partial E}{\partial \theta_j} - \frac{d}{dt}\frac{\partial E}{\partial \dot{\theta}_j} = 0$$



The Optimization Problem

$$\begin{aligned} &\min\ T \\ \text{s.t. } &\frac{dx(t)}{dt} = f(x(t), u(t)) \\ &x(0) = x_0 \\ &x(T) = x_T \\ &0 \le t \le T \end{aligned} \qquad \begin{aligned} &\min\ T \\ \text{s.t. } &\frac{x_{j+1} - x_j}{\Delta\tau} = Tf(x_j, u_j) \\ &x_1 = x_0 \\ &x_n = x_T \\ &1 \le j \le n \end{aligned}$$

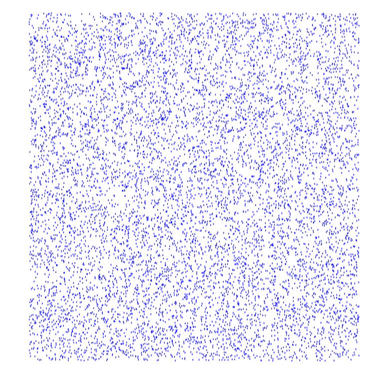
where

- T = total time taken by path
- t = time
- $x(t)$ = state at time t
- $u(t)$ = control at time t
- x_0 = initial state
- x_T = end state

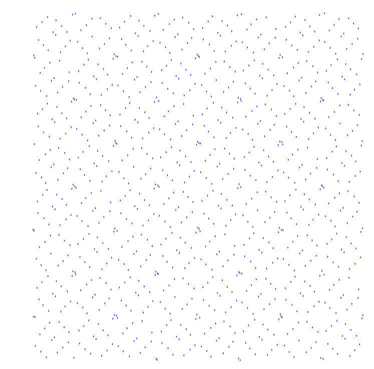
Discretized Precomputation

$$\mathbb{E}[f] = \frac{1}{M}\sum_{i=1}^{M} f(\xi_i)$$

Monte Carlo (MC)

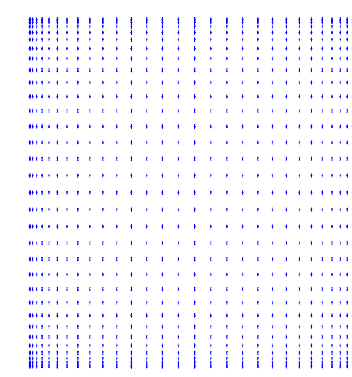


Quasi-Monte Carlo (QMC)

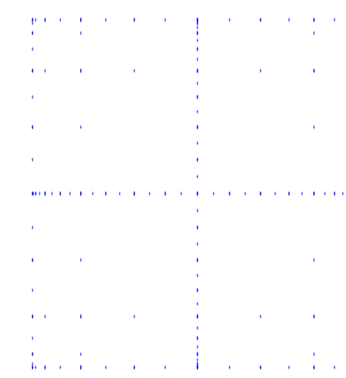


$$\mathbb{E}[f] = \sum_{i=1}^{M} \omega_i f(\xi_i)$$

Tensor Product (TP)



Sparse Grid (SG)



Optimizing the Optimizer

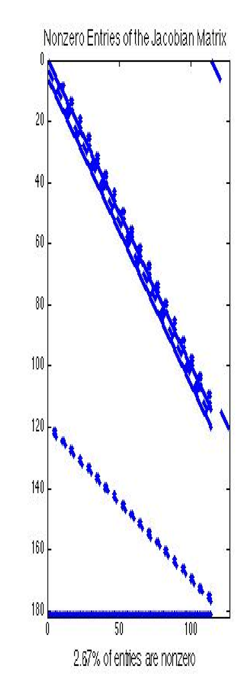
Represent f using hierarchical basis functions (with varying support):

$$f(\xi) \approx \sum_{\mathbf{l}} f_{\mathbf{l}}(\xi), \quad f_{\mathbf{l}}(\xi) = \sum_{\mathbf{i}\ \text{odd}} s_{\mathbf{l},\mathbf{i}}\phi_{\mathbf{l},\mathbf{i}}(\xi)$$

where d-dimensional functions $\phi_{\mathbf{l},\mathbf{i}}(\xi)$ are defined as

$$\phi_{\mathbf{l},\mathbf{i}}(\xi) = \prod_{j=1}^{d} \phi_{l_j,i_j}(\xi_j)$$

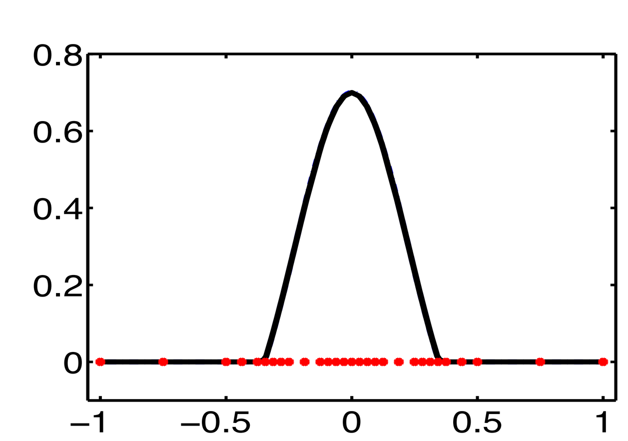
Coefficients $s_{\mathbf{l},\mathbf{i}}$ are called *hierarchical surpluses* as they represent hierarchichal increments between neighboring levels.



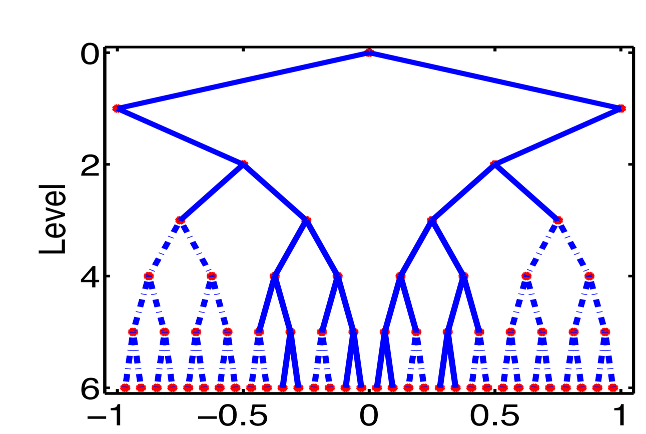
1D hierarchical piecewise linear basis

Optimality

Hierarchical surpluses $s_{\mathbf{l},\mathbf{i}}$ are a natural local error indicator. Thus, refining only points with relatively large surpluses we obtain an adaptive grid (with more points spent in non-smooth areas).

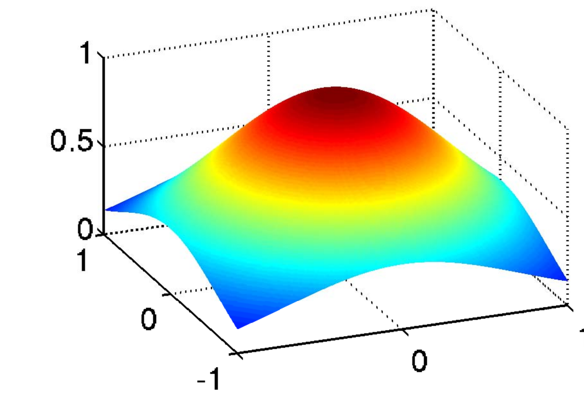


Adaptive grid

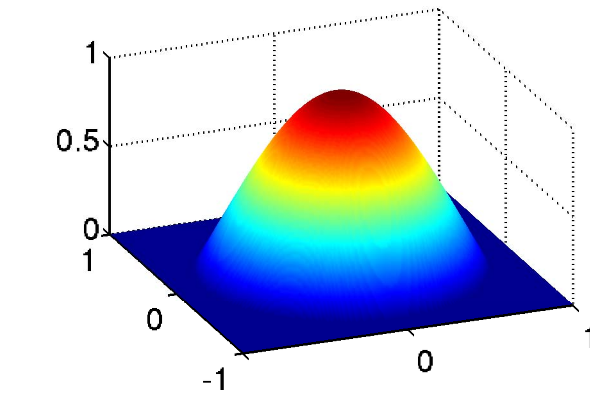


Adaptivity tree

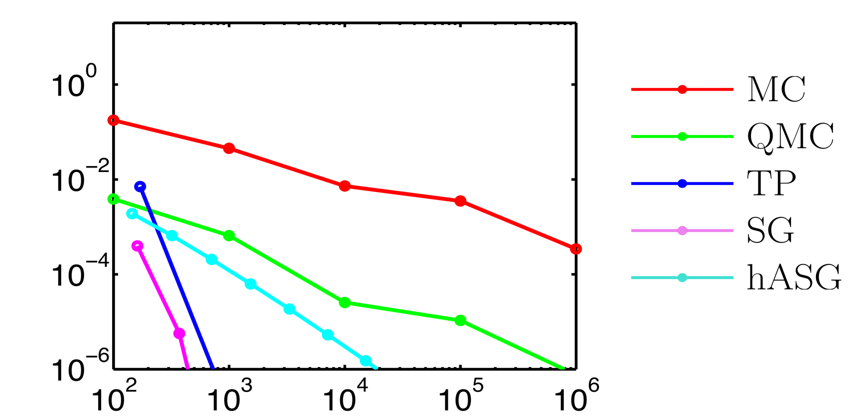
Dog Pics

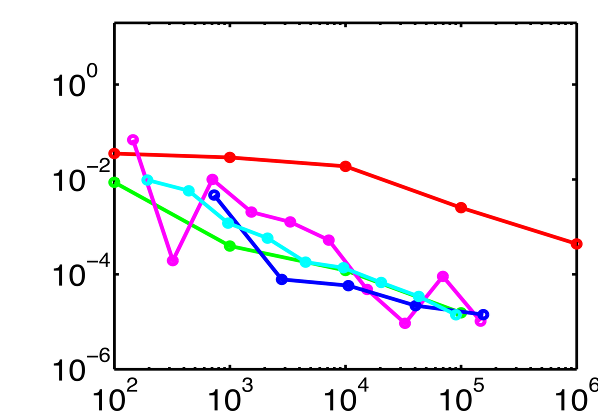


$$f_1(\xi) = \exp\left(-\sum_{i=1}^{d}\xi_i^2\right)$$

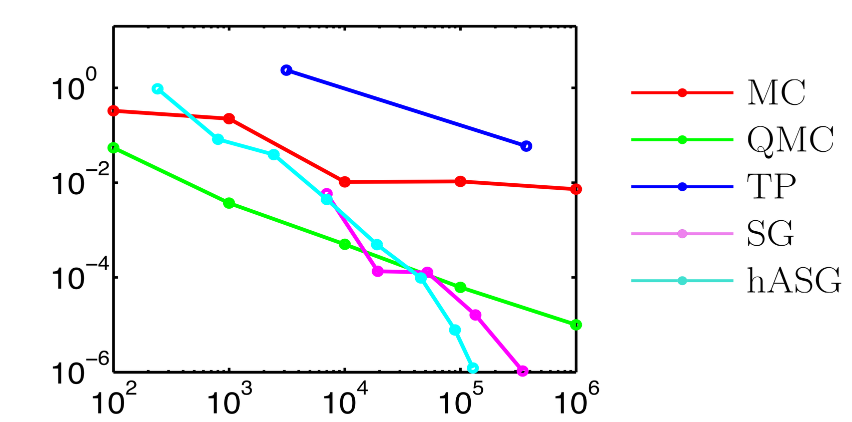


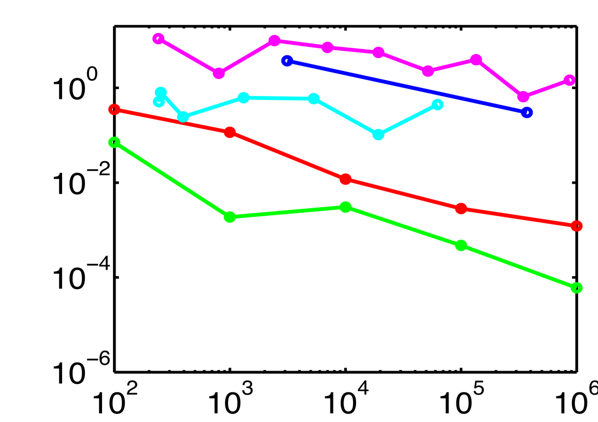
$$f_2(\xi) = 2\max\left\{f_1(\xi) - \tfrac{1}{2}, 0\right\}$$





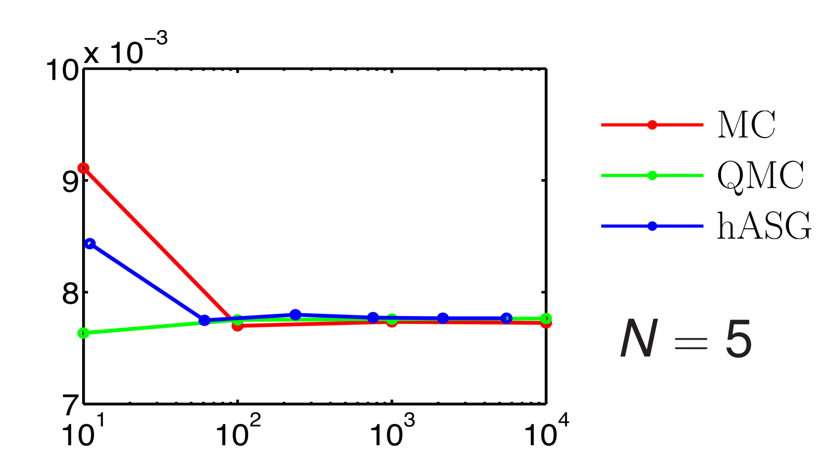
Absolute errors in integral vs the number of samples for $d = 2$

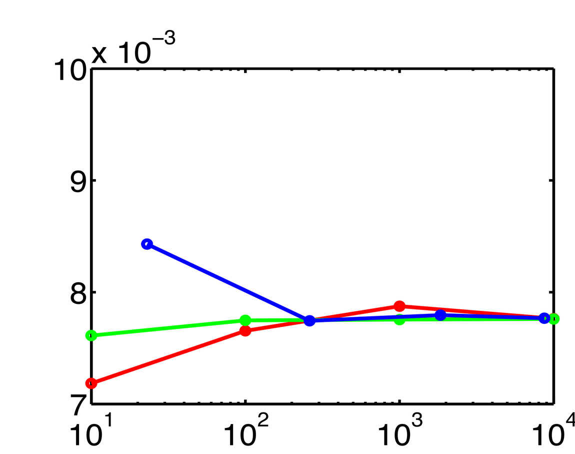




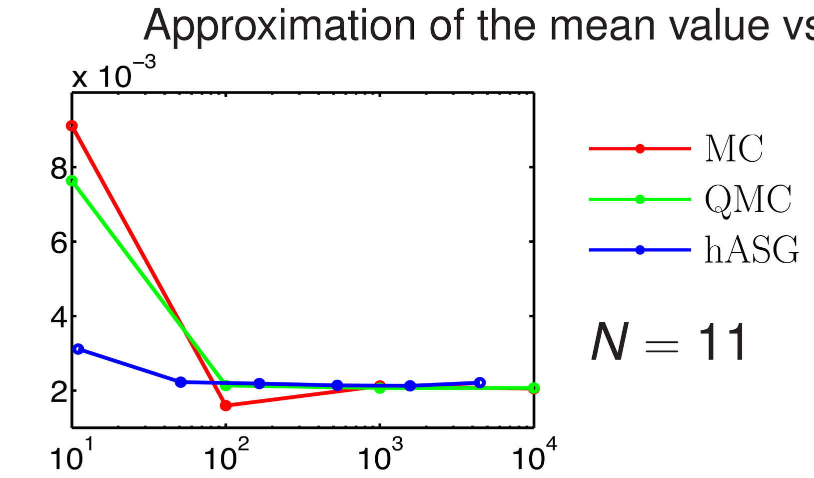
Absolute errors in integral vs the number of samples for $d = 5$

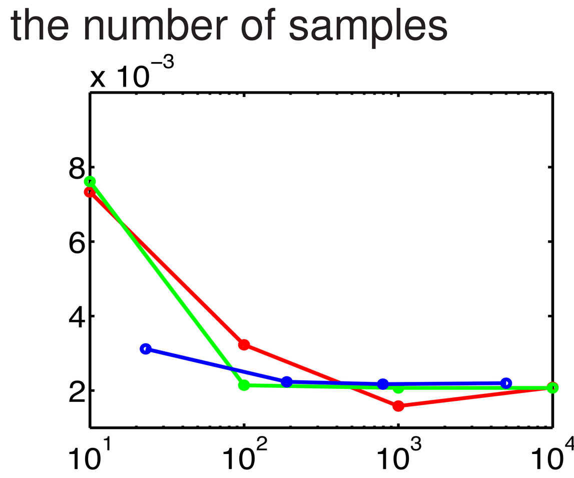
Results





Approximation of the mean value vs the number of samples





Approximation of the semi-deviation vs the number of samples

Future work

- Finding the reason for occasional failure of optimizing with MATLAB's fmincon
- Investigating why the minimum time path is asymmetric from belt to goal and vice versa

Department of Computational and Applied Mathematics

Contact: tat3@rice.edu