Experience with Approximations in the TRPDS Algorithm

Suzanne M. Shontz
Computer Science and Engineering, Penn State
shontz@cse.psu.edu

Joint work with:

Victoria Howle, Patricia Hough
Sandia National Laboratories

Industrial Engineering Seminar
The Pennsylvania State University
April 5, 2007

Outline

- Motivation
- The TRPDS Algorithm
- The TRPDS(p) Algorithm
- Numerical Results
- Case Study: Optimal Design Problem
- Conclusions/Future Work

Motivation

- We are interested in an optimal design problem which is solved by coupling an optimization code and a simulation code.
- The dimension of the optimization problem is small.
- A PDE is solved to evaluate $f(x)$.
- No derivative information is available. The derivative is approximated by computing finite-difference gradients.
- Challenge: High computational expense of the simulation.
- These characteristics are typical of many optimization problems derived from modeling and simulation of physical processes.

The Big Challenge

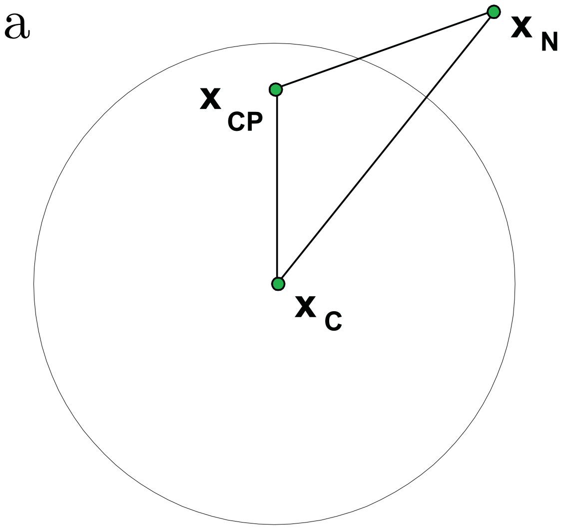
To decrease the amount of time spent on $f(x)$ evaluations.

Combine two approaches:

1. Use parallelism to evaluate functions concurrently.
2. Use approximation models to replace expensive function evaluations with inexpensive model evaluations.

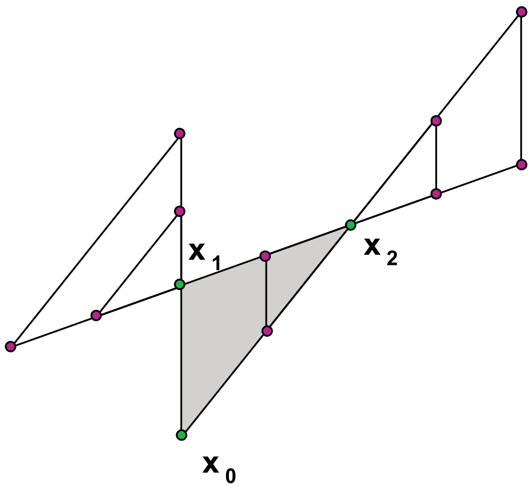
Trust-Region (TR) Method

- **Trust Region:** method that finds the minimum by evaluating the quadratic model in a region where the model is a good approximation to the function
- **Strengths:**
  - Favorable convergence properties
- **Weaknesses:**
  - Uses derivatives that are not always available analytically
  - Computationally expensive function evaluations



Parallel Direct Search (PDS) Method

- **Parallel Direct Search:** parallel method that finds the minimum by evaluating the function at points on a grid.
- **Strengths:**
  - Does not use derivatives
  - Easy to parallelize
- **Weaknesses:**
  - Slow convergence
  - Expensive function evaluations



Trust-Region PDS (TRPDS) Method

- **TRPDS: Combines TR Method and PDS Method**
- **Strengths:**
  - Fast and global convergence
  - Parallelizable
  - Flexibility: Method fits into generalized trust-region framework
- **Weaknesses:**
  - Current implementation does not take advantage of flexibility
  - Current implementation still performs several expensive function evaluations when ratio of sss to processors is large

Algorithm 1. TRPDS Given $\mathbf{x}_0$, $\mathbf{g}_0$, H_0, δ_0, and $\eta \in (0, 1)$

for $k = 0, 1, \ldots$ until convergence **do**

  1. Solve $H_k \mathbf{s}_N = -\mathbf{g}_k$

  for $i = 0, 1, \ldots$ until step accepted **do**

    2. Form an initial simplex using $\mathbf{s}_N$

    3. Find an approximate solution $\mathbf{s}_i$ that minimizes $f(\mathbf{x}_k + \mathbf{s})$

    if $ared/pred > \eta$ **then**

      4. Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_i$; Evaluate $\mathbf{g}_{k+1}$ and H_{k+1}

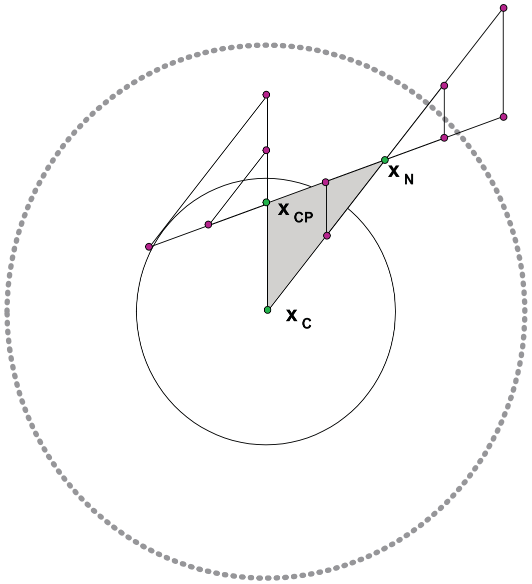
    end if

    5. Update δ

  end for

end for

An Overview of TRPDS



Generalized Trust-Region Framework

The framework is for managing the use of approximation models. (Alexandrov, Dennis, Lewis, and Torczon, 1998)

- An *approximation model* $a_k(\mathbf{x}_k)$ is a less expensive representation of $f(\mathbf{x}_k)$.
- The trust-region method with generalized approximation models converges globally whenever:
  1. $a_k(\mathbf{x}_k) = f(\mathbf{x}_k)$
  2. $\nabla a_k(\mathbf{x}_k) = \nabla f(\mathbf{x}_k)$
- Steps can be computed in any manner as long as the sequence of iterates produced satisfies the fraction of Cauchy decrease (FCD) condition.

TRPDS with Generalized Approximation Models

The model management framework is employed as follows:

- At iteration k, an approximation model, $m_k(\mathbf{x}_k)$, to the objective function, $f(\mathbf{x}_k)$, is built.
- Then, the following PDS subproblem is solved approximately:

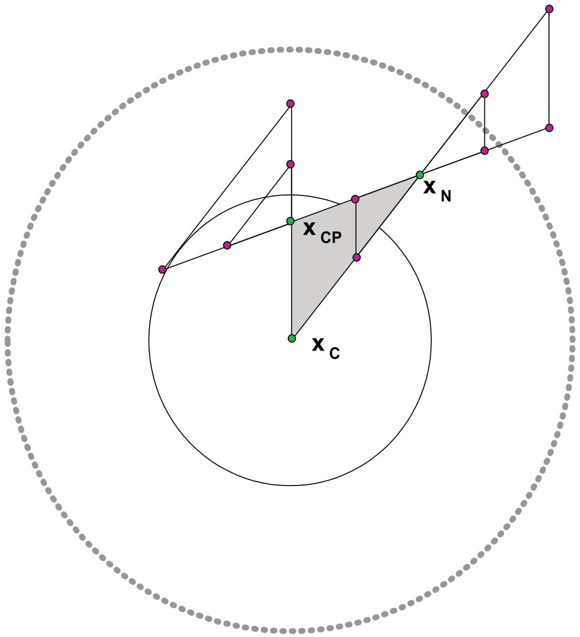
$$\begin{aligned} &\min m_k(\mathbf{x}_k + \mathbf{s}) \\ &s.t. \ \|\mathbf{s}\|_2 \leq 2\delta_k \end{aligned}$$

- Our approximation model, $m_k(\mathbf{x}_k)$, is used only to solve the PDS subproblem. We model $f(\mathbf{x}_k)$ by a quadratic model, i.e., $a_k(\mathbf{x}_k + \mathbf{s}) = f(\mathbf{x}_k) + \mathbf{g}_k^T \mathbf{s} + \frac{1}{2}\mathbf{s}^T H_k \mathbf{s}$, when determining if FCD has been satisfied.

TRPDS(p): Incorporating Approximation Models

Algorithm 2. TRPDS(p) (Howle, Shontz, and Hough, 2000)
Given p processors, m_0, $\mathbf{x}_0$, $\mathbf{g}_0$, H_0, δ_0, and $\eta \in (0,1)$
for $k = 0, 1, \ldots$ until convergence **do**
  1. Solve $H_k \mathbf{s}_N = -\mathbf{g}_k$
  for $i = 0, 1, \ldots$ until step accepted **do**
    2. Form initial simplex using $\mathbf{s}_N$
    3. Compute p best approximate solutions $\mathbf{s}_1, \ldots, \mathbf{s}_p$ that minimize $m_k(\mathbf{x}_k + \mathbf{s})$
    4. Compute $\mathbf{s}_i$ that minimizes $f(\mathbf{x}_k + \mathbf{s})$
    if $ared/pred > \eta$ **then**
      5. Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_i$; Evaluate $\mathbf{g}_{k+1}$, $\mathbf{H}_{k+1}$
    end if
    6. Update δ
  end for
end for

An Overview of TRPDS(p)



The Use of Quadratic Models in TRPDS(p)

- Initially, we implemented TRPDS(p) for the case where m_k is a quadratic model, i.e.,

$$m_k(\mathbf{x}_k + \mathbf{s}) = f(\mathbf{x}_k) + \mathbf{g}_k^T \mathbf{s} + \frac{1}{2}\mathbf{s}^T H_k \mathbf{s}.$$

- Initial numerical results (on standard test problems) correspond to this case.

- Later, TRPDS(p) was extended to include the use of more general approximation models.

Standard Test Problems

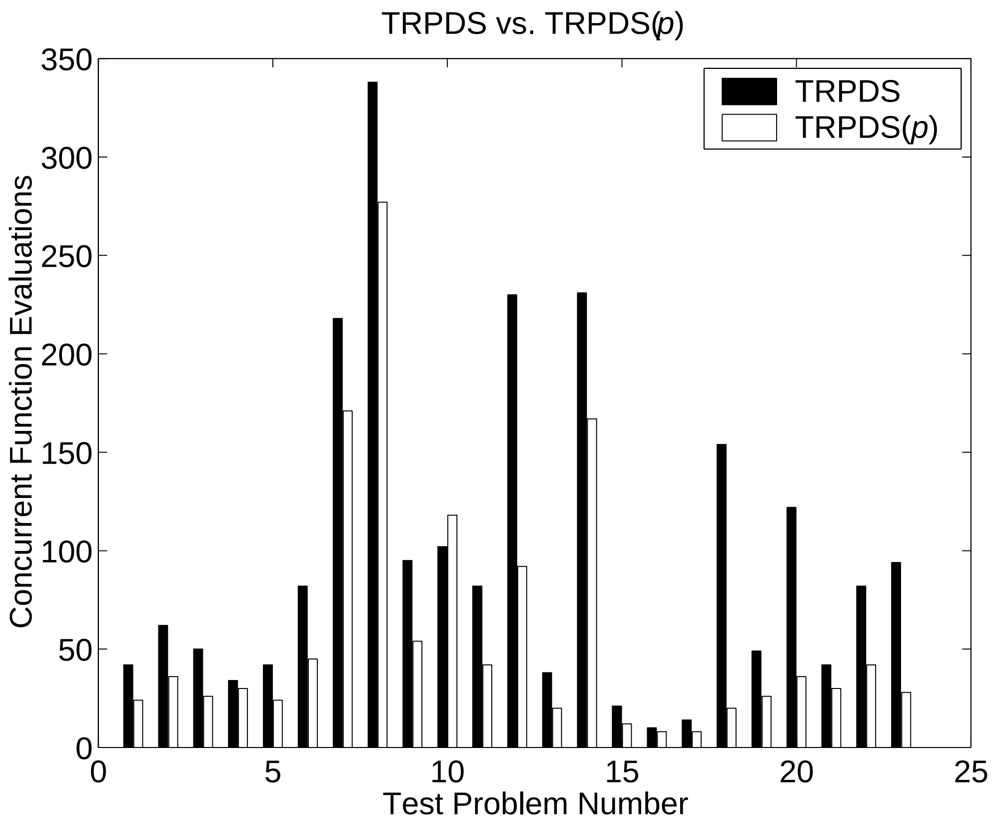
To compare the performance of TRPDS and TRPDS(p), we solved a standard set of test problems from papers by Moré, Garbow, and Hillstrom (Moré, et. al., 1981), Byrd, Schnabel, and Shultz (Byrd, et. al., 1988), and Conn, Gould, and Toint (Conn, et. al., 1986).

- The starting points used for these problems were the same as those given in the references.
- We used the step tolerance, the function tolerance, and the gradient tolerance as stopping conditions.
- We recorded the number of concurrent function evaluations for comparisons.

Initial Conditions for Test Problems

$$
\begin{aligned}
\text{Machine Epsilon} &= 2.22045 \times 10^{-16} \\
\text{Maximum Step} &= 1000 \\
\text{Minimum Step} &= 1.49012 \times 10^{-8} \\
\text{Maximum Iter} &= 500 \\
\text{Maximum Fcn Eval} &= 10000 \\
\text{Step Tolerance} &= 1.49012 \times 10^{-8} \\
\text{Function Tolerance} &= 1.49012 \times 10^{-8} \\
\text{Gradient Tolerance} &= 6.05545 \times 10^{-6} \\
\text{LineSearch Tolerance} &= 0.0001.
\end{aligned}
$$

Test Problem Results: dim=8, p=9

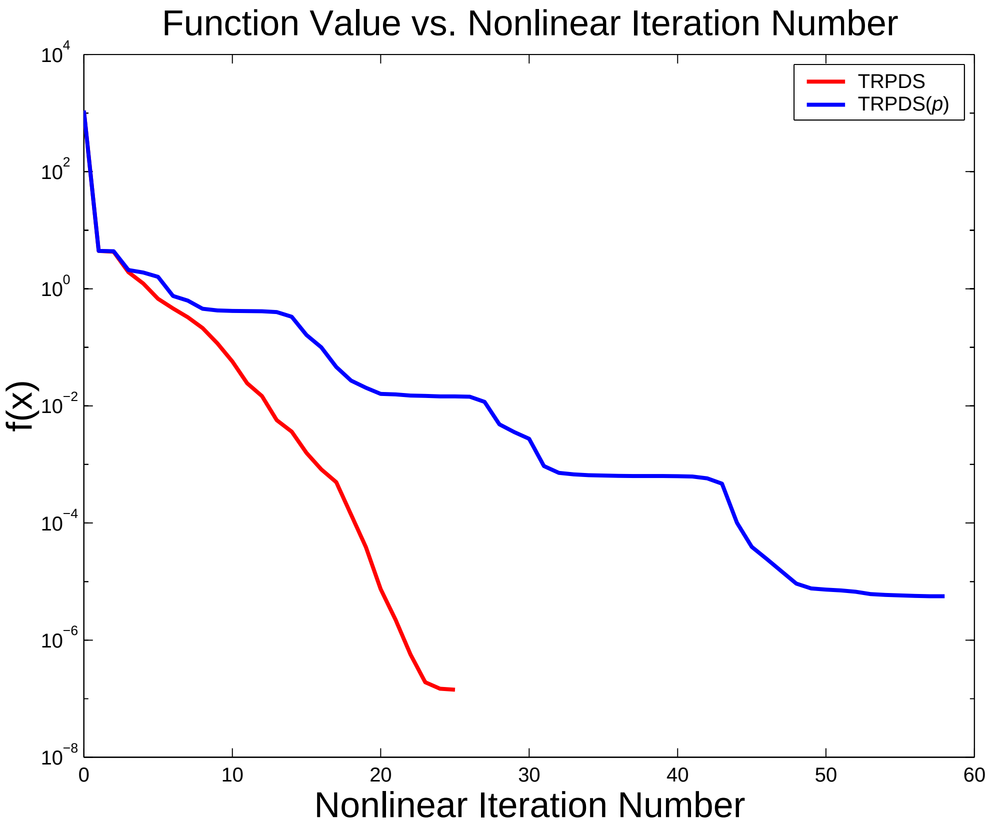


TRPDS(p) with a quadratic model nearly always beats TRPDS.

Problem 10: cragg_levy

- TRPDS(p) required 118 concurrent fevals and 58 iterations.
- TRPDS required 102 concurrent fevals and 25 iterations.
- Because the problem dimension must be divisible by 4, no contour plots can be drawn.
- The quadratic model in TRPDS(p) often predicts increase and is thus not a good choice of approximation model for this problem.

Problem 10: cragg_levy



The quadratic model in TRPDS(p) often predicts increase.

TRPDS(p) Statistics

Over all test problems:

- TRPDS(p) with QM beats TRPDS 95 percent of the time with respect to concurrent function evaluations.

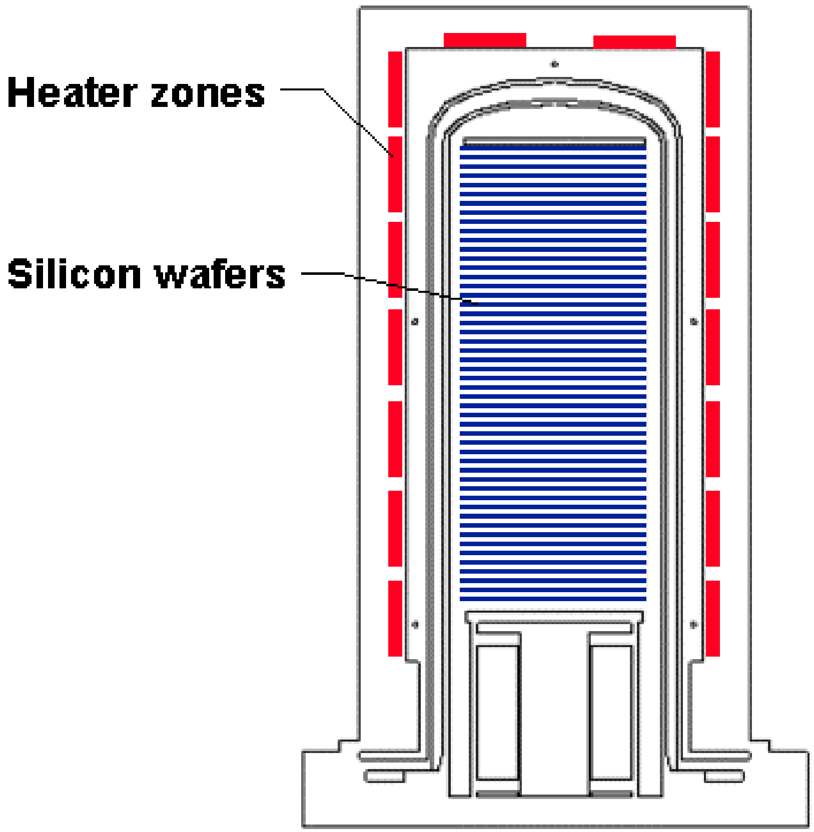
- TRPDS(p) with QM yielded up to 87 percent improvement over TRPDS with respect to concurrent function evaluations.

- TRPDS(p) with QM yields an average improvement of 36 percent over TRPDS with respect to concurrent function evaluations.

Using Generalized Approximation Models in TRPDS(p)

- We extended our implementation of TRPDS(p) to include the use of a general approximation model within the PDS portion of TRPDS(p).
- Now, any approximation model (physical, numerical, mathematical, etc.) can be used to approximately solve the PDS subproblem.
- A quadratic model is still used to determine if the FCD condition has been satisfied. This could be made more general.

Case Study: Optimal Design Problem



- Chemical vapor deposition reactor
- Heating coils controlled by zone
- Goal: Uniform chemical deposition on each of the wafers

TWAFER Optimization Problem

- The power settings, p_j, are optimized to achieve a particular uniform temperature (with the goal of achieving uniform chemical deposition on each wafer).
- The objective function, f, is defined by a least-squares fit of the N discrete wafer temperatures, T_i, to a prescribed temperature, T^*,

$$f\left(\mathbf{p}\right) = \sum_{i=1}^{N} \left(T_i - T^*\right)^2$$

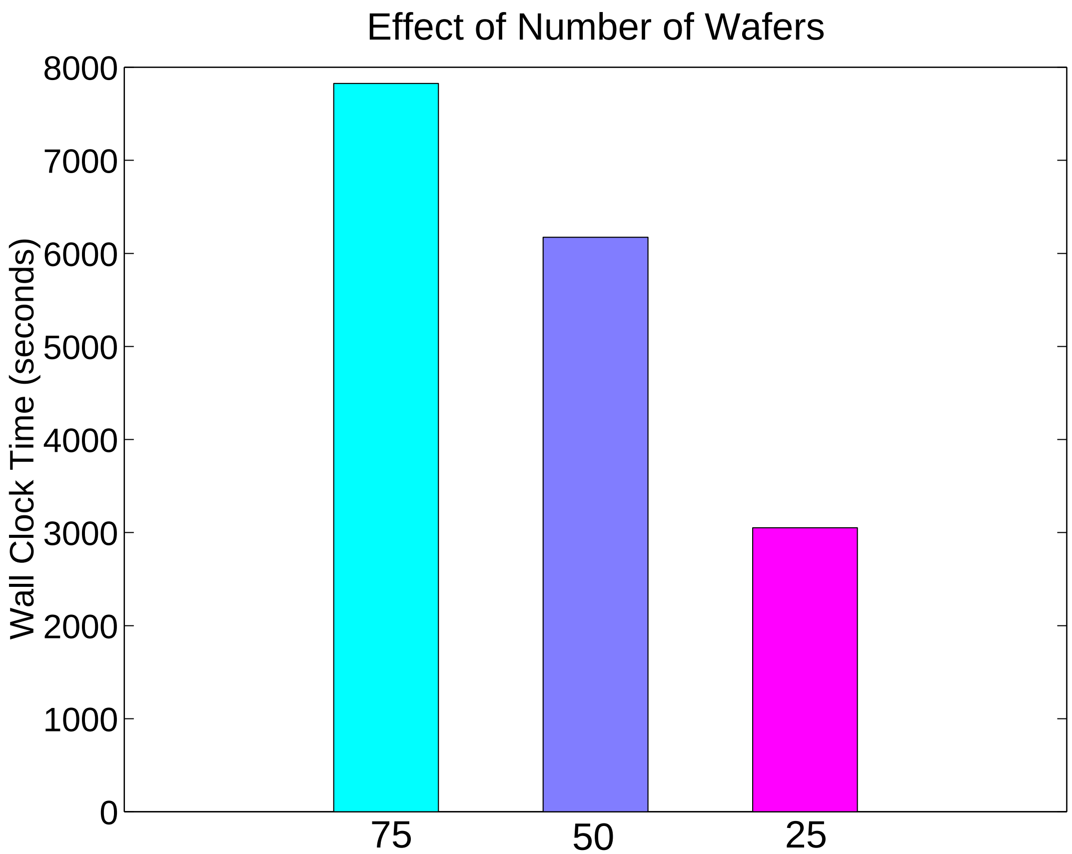
where the p_j are the unknown power parameters.

Approximation Models

We implemented several approximation models for the optimal design problem including the following:

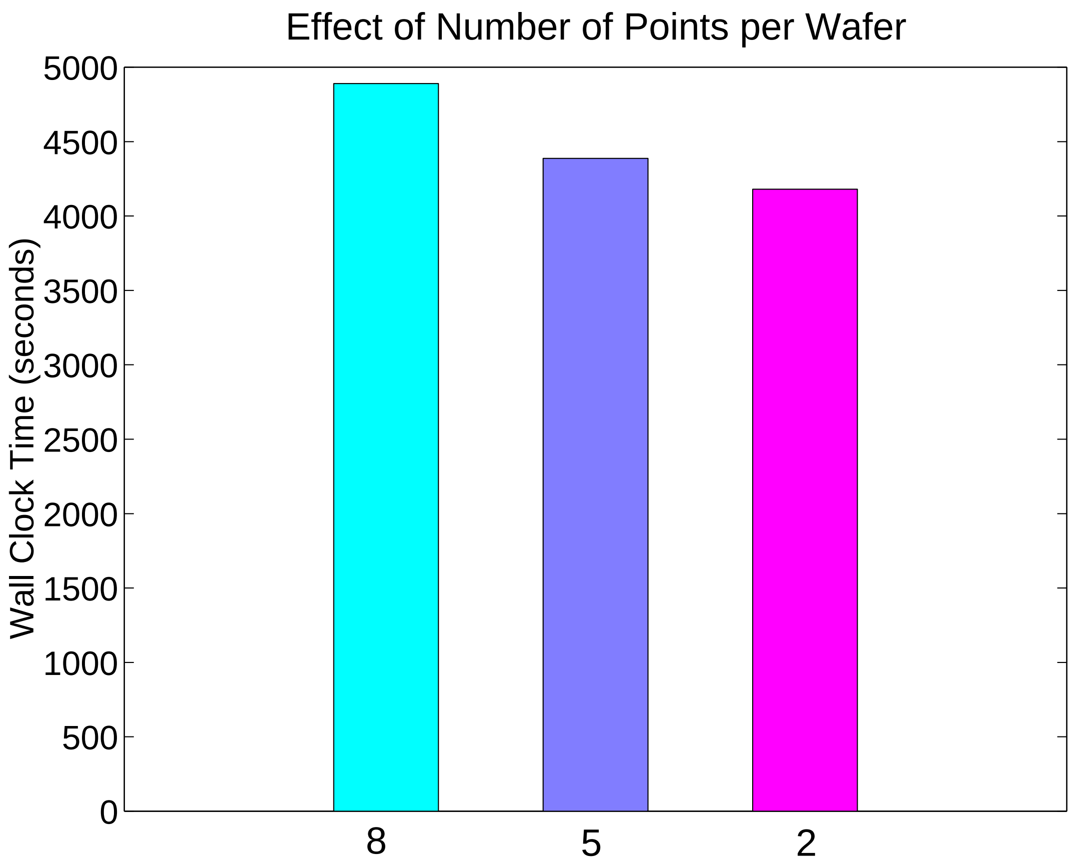
- Mathematical Model:
  - Quadratic Model
- Discretization Models:
  - Coarse-to-fine discretization: Number of wafers
  - Coarse-to-fine discretization: Number of points per wafer
  - Coarse-to-fine discretization: Based on both
- Numerical Model:
  - Loose-to-Tight PDE Convergence Tolerances

Twafer Results: $dim = 7$, $p = 8$, $sss = 56$



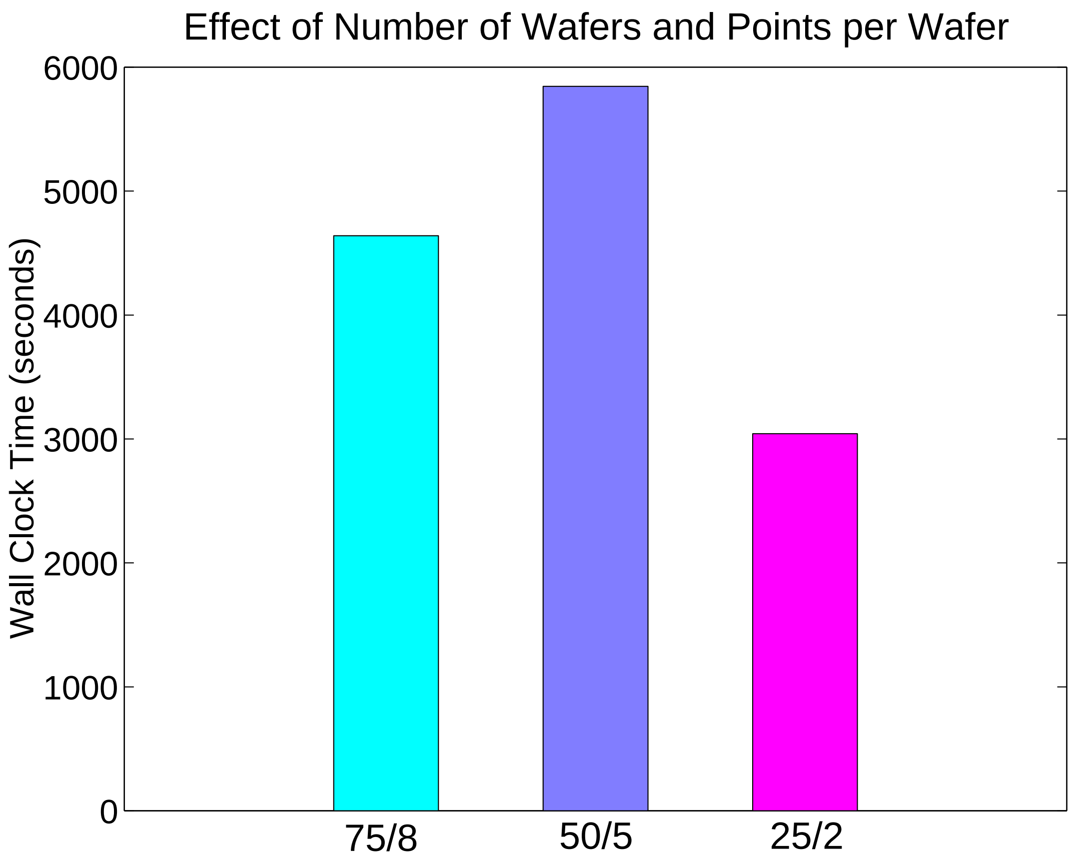
TRPDS(p) decreases solution time by decreasing the number of wafers.

Twafer Results: $dim = \mathbf{7}$, $p = \mathbf{8}$, $sss = \mathbf{56}$



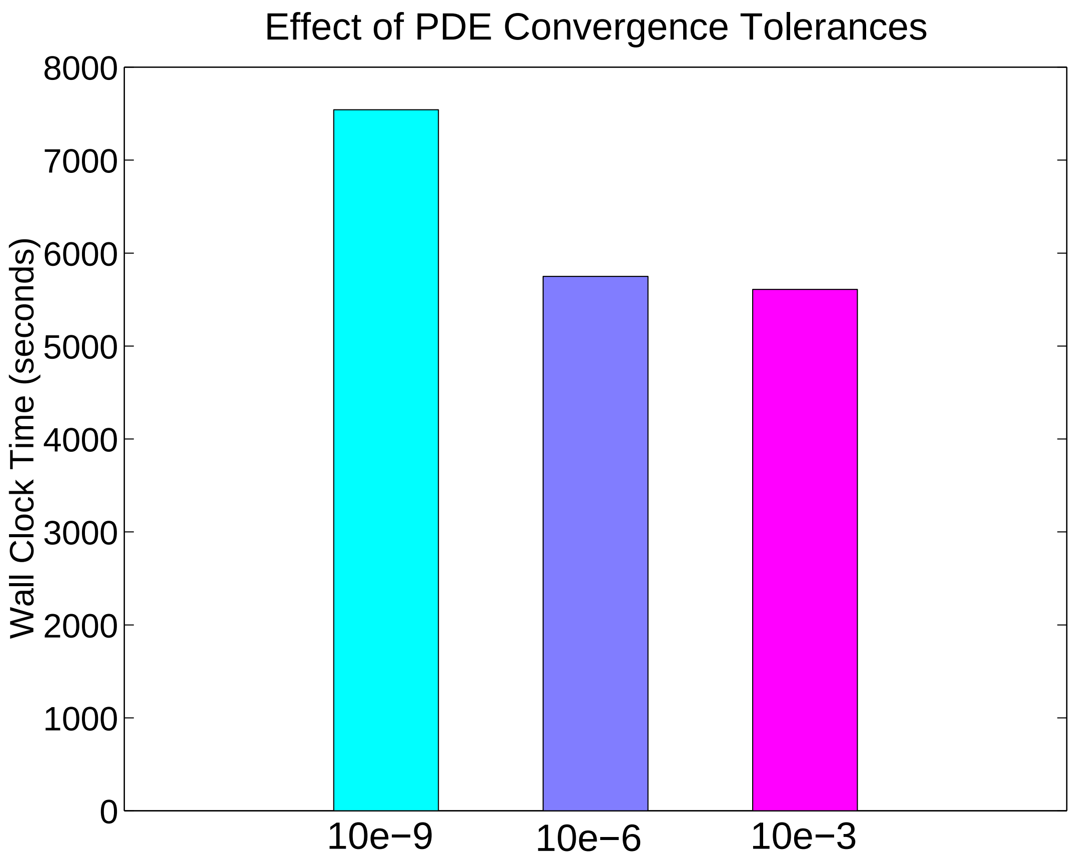
TRPDS(p) decreases solution time by decreasing the number of points per wafer.

Twafer Results: $dim = \mathbf{7}$, $p = \mathbf{8}$, $sss = \mathbf{56}$



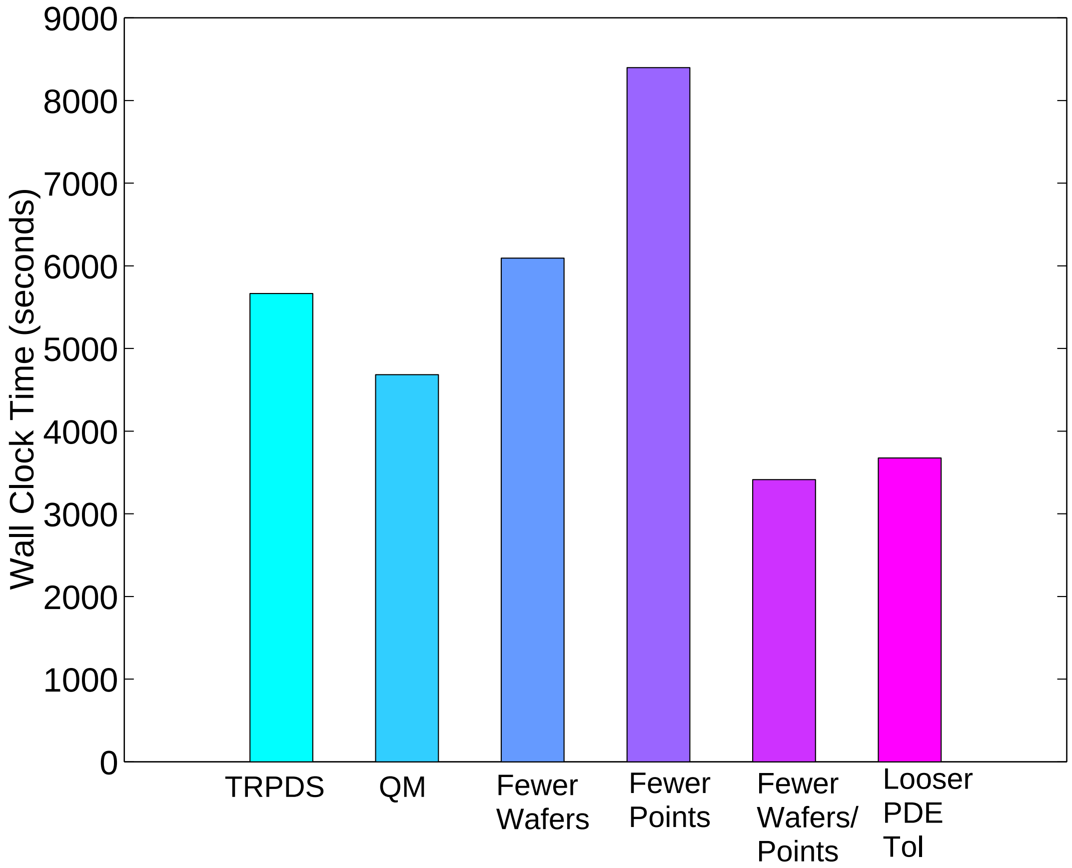
It's harder to choose the appropriate number of wafers and points per wafer for TRPDS(p).

Twafer Results: $dim = \mathbf{7}$, $p = \mathbf{8}$, $sss = \mathbf{56}$



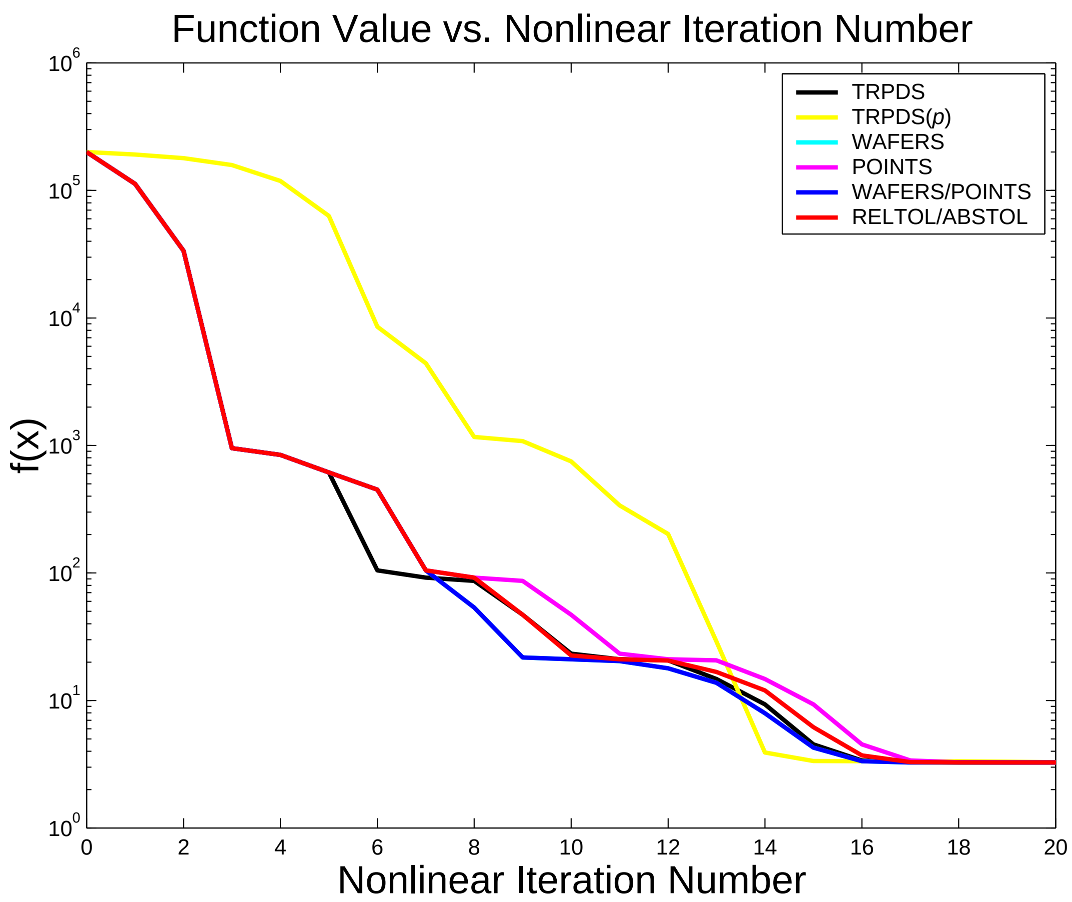
TRPDS(p) decreases solution time by making convergence tolerances looser.

Twafer Results: dim = 7, $p = 14$, $sss = 14$



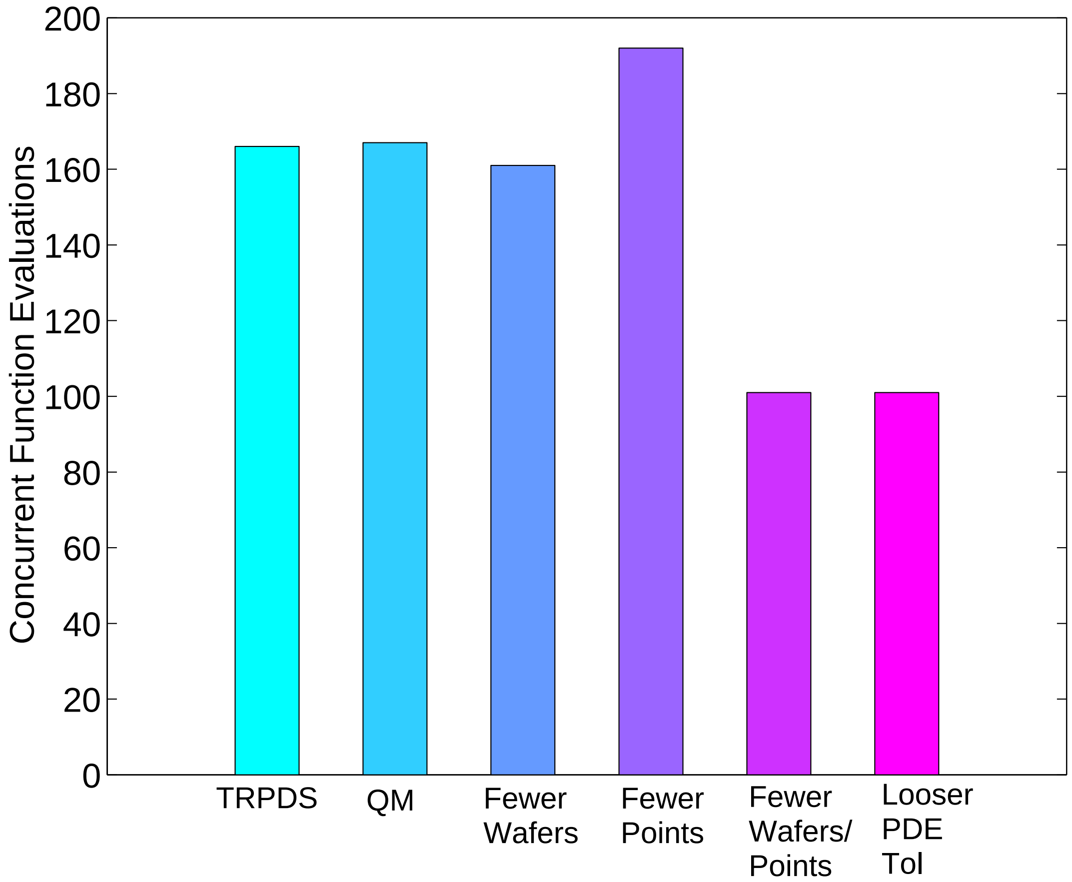
TRPDS(p) decreases solution time with the use of appropriate approximation models.

Twafer Results: dim = 7, p = 14, sss=14



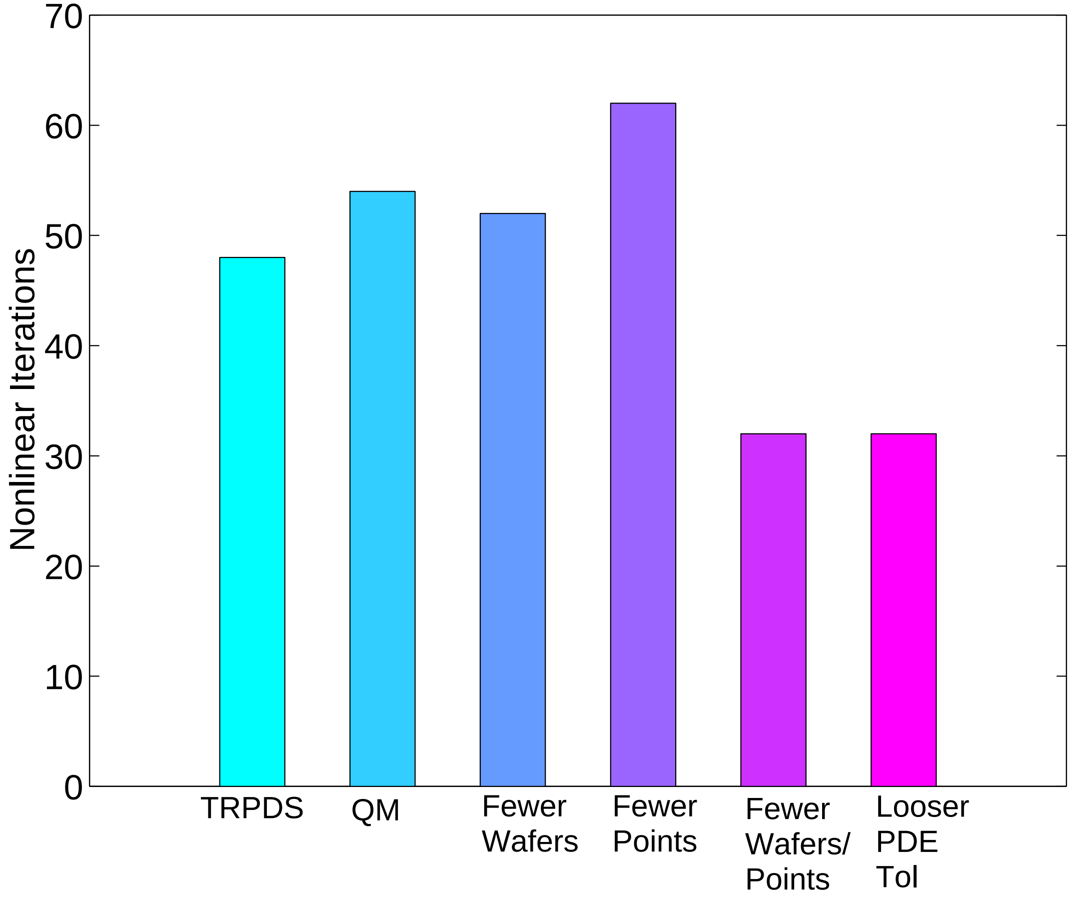
Behavior of TRPDS(p) using various approximation models.

Twafer Results: $dim = \mathbf{7}$, $p = \mathbf{14}$, sss=**14**



TRPDS(p) reduces concurrent function evaluations.

Twafer Results: $dim = 7$, $p = 14$, $sss = 14$



TRPDS(p) reduces nonlinear iterations.

Twafer Results: $dim = \mathbf{7}$, $p = \mathbf{8}$, $sss = \mathbf{56}$

Model Timing Rankings:

1. WAFERS25/POINTS2
2. WAFERS25
3. RTOL3/ATOL6
4. WAFERS50/POINTS5
5. WAFERS50
6. POINTS2
7. RTOL6/ATOL9
8. WAFERS75/POINTS8
9. POINTS5
10. WAFERS75
11. RTOL9/ATOL12
12. POINTS8
13. f

Optimization Rankings:

1. WAFERS25/POINTS2
2. WAFERS25
3. POINTS2
4. POINTS5
5. WAFERS75/POINTS8
6. QM
7. POINTS8
8. RTOL3/ATOL6
9. RTOL6/ATOL9
10. WAFERS50/POINTS5
11. WAFERS50
12. RTOL9/ATOL12
13. WAFERS75

Conclusions

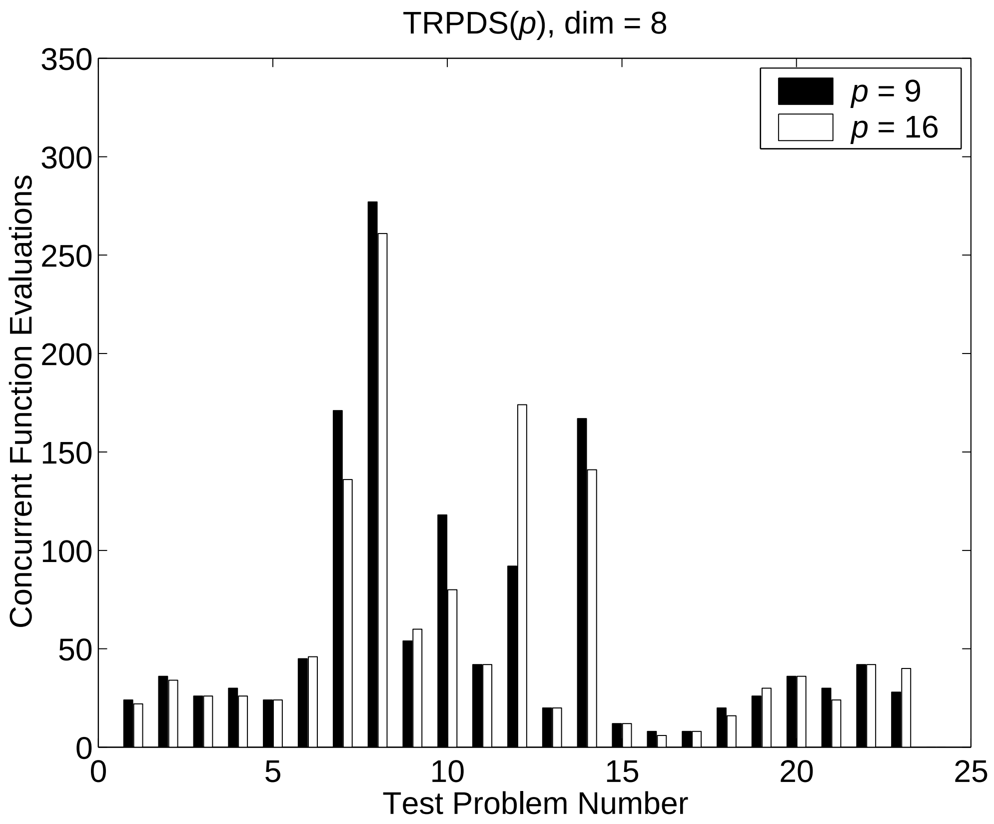
- Added approximation model option in TRPDS(p) in solution of the PDS subproblem
- Use of approximation models in TRPDS(p) was competitive on an engineering application
- Most successful models were less expensive models where TRPDS(p) spent less time on solution of PDS subproblem.
- Important to consider tradeoffs between model cost and fidelity.

Future Work

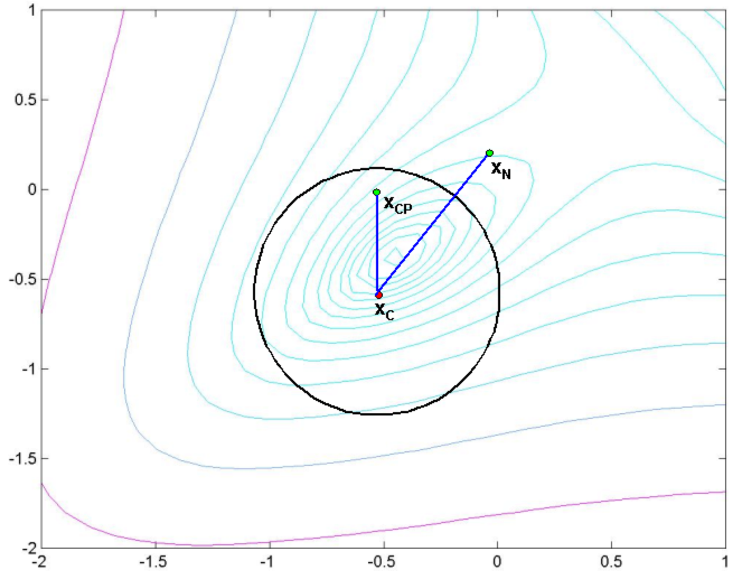
- Develop a range of model strategies within PDS subproblem
- Replace quadratic model with other approximation at the trust region level
- Incorporate a speculative gradient capability in TRPDS(p)
- Evaluate with forthcoming GSS variation (Meza, Oliva)

Motivation for Speculative Gradients:

Inefficient Use of Extra Parallelism



An Iteration of Speculative Gradient



1. Minimize quadratic model over trust region
2. Processor 0 evaluates trial iterate;
   Remaining processors evaluate gradient
3. Check sufficient decrease
4. Accept/Reject trial iterate
5. Update trust region
6. Goto 1

Results

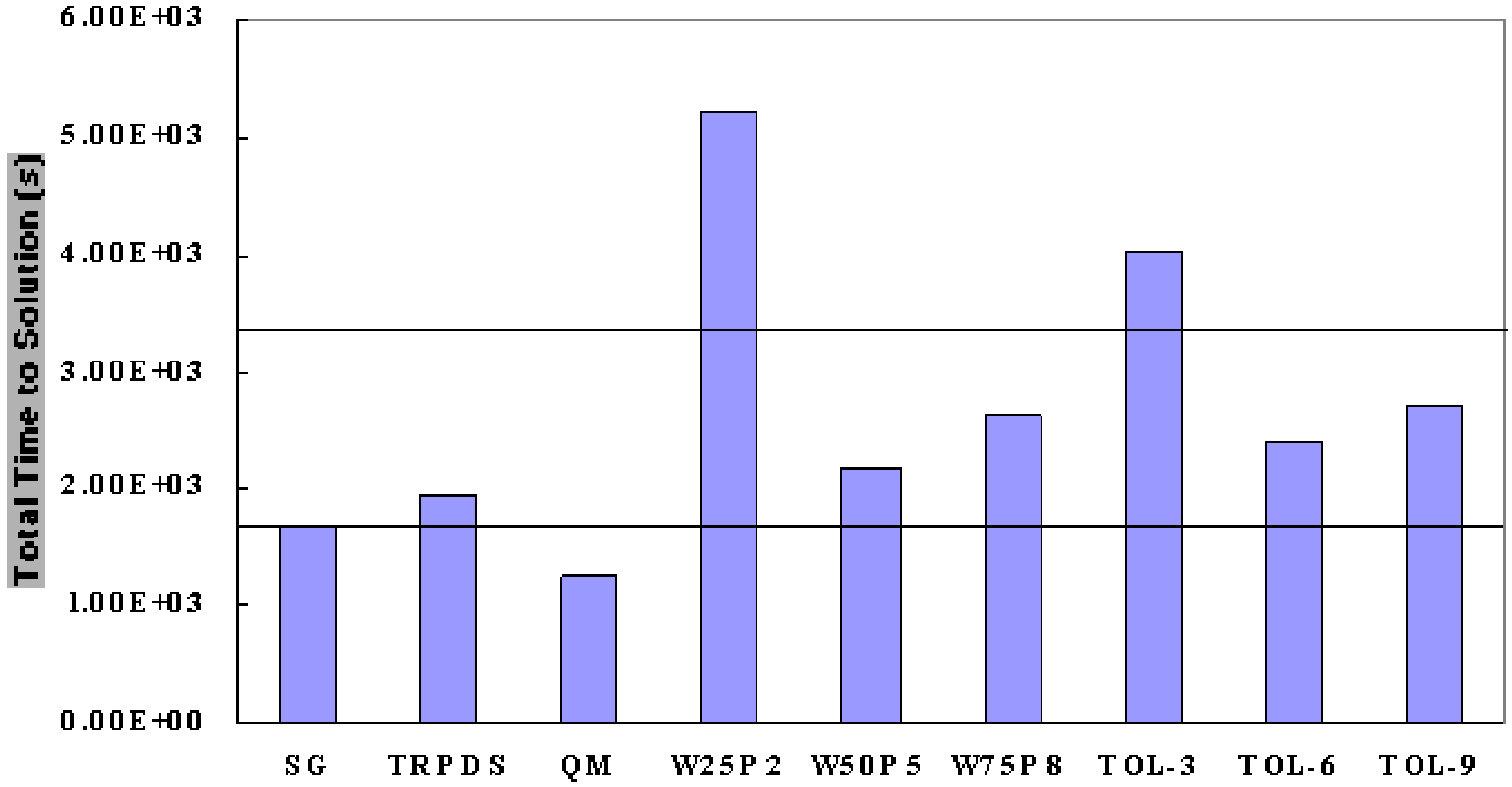


Figure 1: QuadraticModel < SpecGrad < TRPDS.

Most models are within a factor of two of SpecGrad.

Results

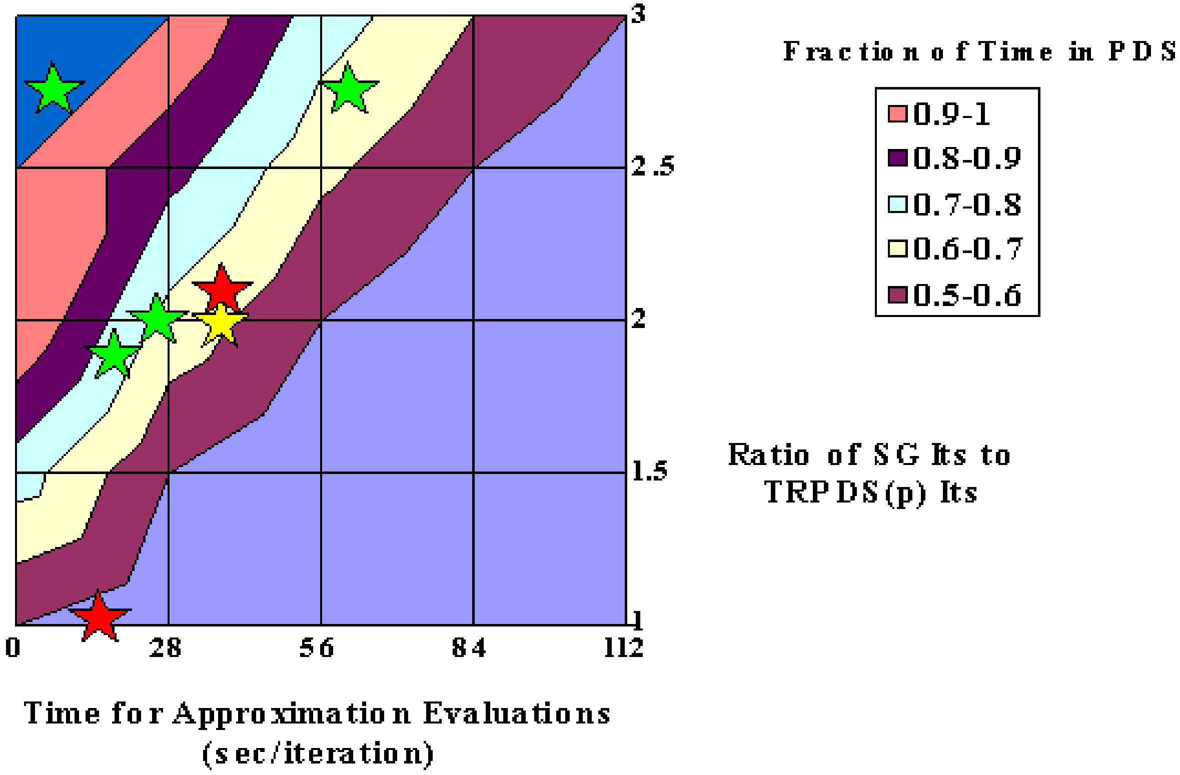
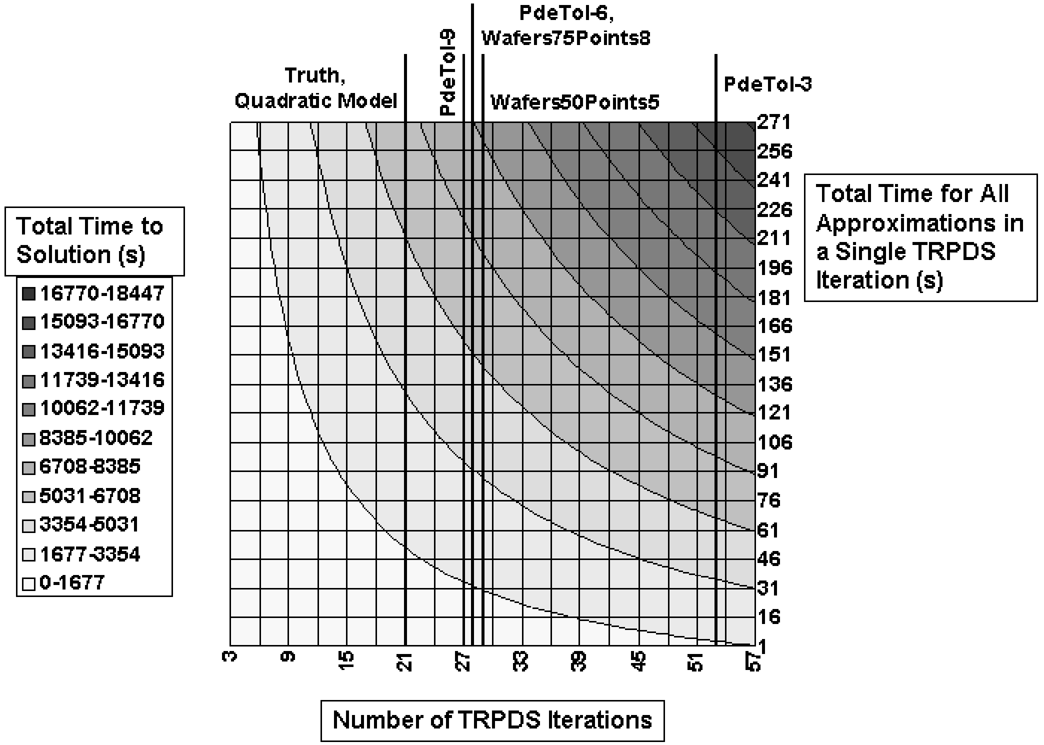


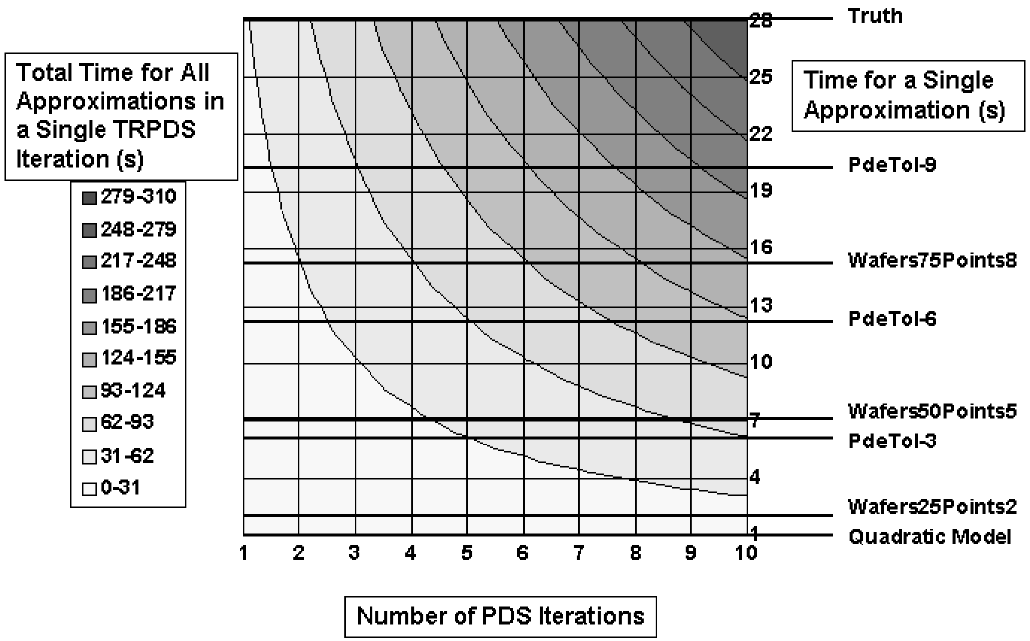
Figure 2: Contour plot illustrates relationship that demonstrates TRPDS(p) with a speculative strategy would have done better than speculative gradient.

Total Time to Solution



Contour plot demonstrating tradeoff between model cost and fidelity.

Time Spent on Approximation Evaluations



The time required for approximation evals can quickly build up with the number of PDS iterations for more expensive approximations.