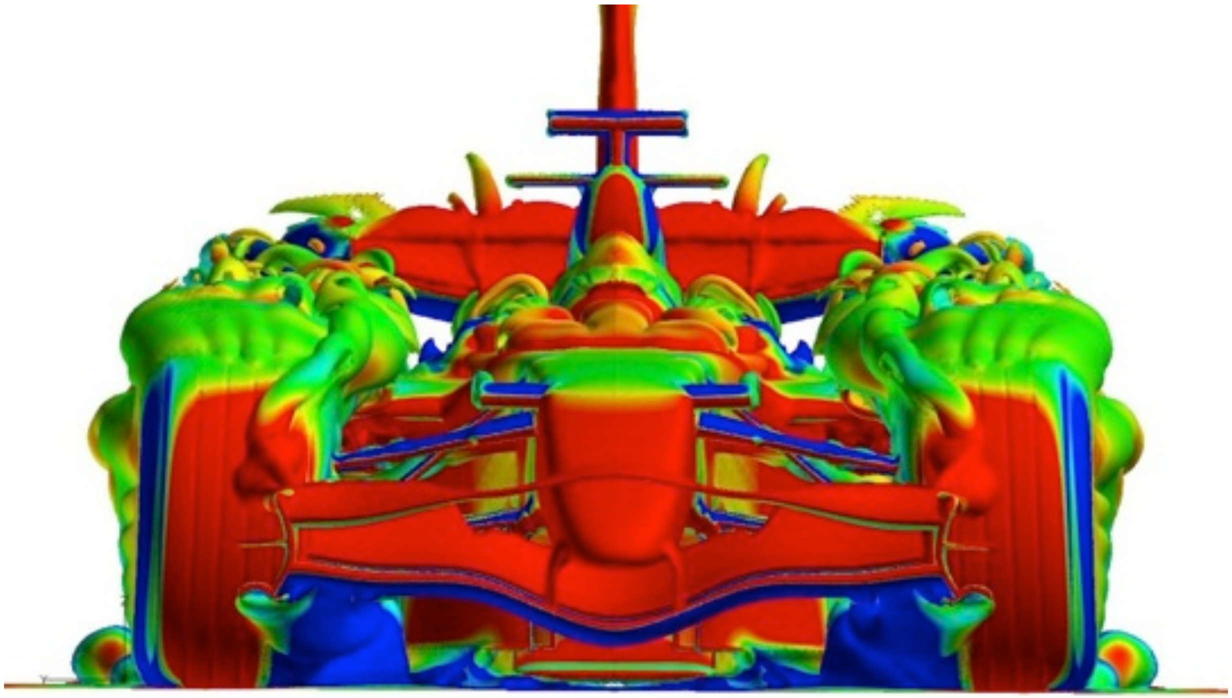
Enforcing Boundary Conditions for Reduced-Order CFD Simulations

Kyle Washabaugh
Farhat Research Group, Stanford University

CME334 Presentation
Stanford, CA
October 24, 2013

Motivation for ROMs

Problem: High-fidelity CFD simulations are too computationally expensive for time sensitive applications.



Goal: Reduce computational complexity *without* coarsening model or omitting relevant physics.

Projection-Based Model Reduction

Consider a set of ODEs arising from the discretization in space of a space-time PDE:

$$\frac{d\mathbf{w}(t)}{dt} = \mathbf{f}(\mathbf{w}(t), t, \mu)$$
$$\mathbf{w}(0) = \mathbf{w}_0,$$

where $t \geq 0$ denotes time, $\mathbf{w}(t) \in \mathbb{R}^n$ denotes the fluid state vector, and $\mu \in \mathbb{R}^d$ denotes a vector of parameters defining the operating point.

Projection-Based Model Reduction

Using implicit time integration, the state $\mathbf{w}^{(i)} \in \mathbb{R}^n$ at time $t^{(i)}, 0 \leq i \leq N_t$ can be computed as the solution to a discrete nonlinear residual

$$\mathbf{r}^{(i)}(\mathbf{w}^{(i)}, \mu) = \mathbf{0}.$$

Projection-Based Model Reduction

Using implicit time integration, the state $\mathbf{w}^{(i)} \in \mathbb{R}^n$ at time $t^{(i)}, 0 \leq i \leq N_t$ can be computed as the solution to a discrete nonlinear residual

$$\mathbf{r}^{(i)}(\mathbf{w}^{(i)}, \mu) = \mathbf{0}.$$

Introducing a reduced order basis (ROB), $\mathbf{V} \in \mathbb{R}^{n \times k}$, leads to a system of n eq. for $k \ll n$ variables

$$\min_{\Delta \mathbf{w}_k^{(i)} \in \mathbb{R}^k} \left\| \mathbf{r}^{(i)}(\mathbf{w}^{(i-1)} + \mathbf{V} \Delta \mathbf{w}_k^{(i)}, \mu) \right\|_2^2$$

Minimum-Residual Based ROM

Bui-Tanh et al. 2008, Carlberg et al. 2011

Projection-Based Model Reduction

Using implicit time integration, the state $\mathbf{w}^{(i)} \in \mathbb{R}^n$ at time $t^{(i)}, 0 \leq i \leq N_t$ can be computed as the solution to a discrete nonlinear residual

$$\mathbf{r}^{(i)}(\mathbf{w}^{(i)}, \mu) = \mathbf{0}.$$

Introducing a reduced order basis (ROB), $\mathbf{V} \in \mathbb{R}^{n \times k}$, leads to a system of n eq. for $k \ll n$ variables

$$\min_{\Delta \mathbf{w}_k^{(i)} \in \mathbb{R}^k} \left\| \mathbf{r}^{(i)}(\mathbf{w}^{(i-1)} + \mathbf{V}\Delta\mathbf{w}_k^{(i)}, \mu) \right\|_2^2$$

420x reduction in CPU with less than 1% error in outputs
GNAT method: Carlberg et al. 2011

Enforcing Boundary Conditions

Partitioning the residual into interior / boundary:

$$\begin{bmatrix} \mathbf{r}^{(i)}_{\text{interior}}(\mathbf{w}^{(i)}, \mu) \\ \\ \mathbf{r}^{(i)}_{\text{boundary}}(\mathbf{w}^{(i)}, \mu) \end{bmatrix} = \mathbf{0}$$

Enforcing Boundary Conditions

Partitioning the residual into interior / boundary:

$$\begin{bmatrix} \mathbf{r}^{(i)}_{\text{interior}}(\mathbf{w}^{(i)}, \mu) \\ \\ \mathbf{r}^{(i)}_{\text{boundary}}(\mathbf{w}^{(i)}, \mu) \end{bmatrix} = \mathbf{0}$$

If any basis vectors do not satisfy the boundary conditions, then the relative weighting of these terms becomes important in the minimization:

$$\min_{\Delta \mathbf{w}^{(i)}_k \in \mathbb{R}^k} \left\| \begin{bmatrix} \mathbf{r}^{(i)}_{\text{interior}}(\mathbf{w}^{(i-1)} + \mathbf{V}\Delta\mathbf{w}^{(i)}_k, \mu) \\ \\ \epsilon(\mathbf{r}^{(i)}_{\text{boundary}}(\mathbf{w}^{(i-1)} + \mathbf{V}\Delta\mathbf{w}^{(i)}_k, \mu)) \end{bmatrix} \right\|^2_2$$

Gauss Newton Iterations

In practice, this weighted nonlinear least squares problem can be solved using the Gauss-Newton Algorithm, resulting in the following iterations:

$$\min_{\Delta \mathbf{w}^{(i)} \in \mathbb{R}^n} \left\| \mathbf{J}^{(i,l-1)} \mathbf{V} \Delta \mathbf{w}_k^{(i,l)} + \mathbf{r}^{(i,l-1)} \right\|_{\mathbf{P}}^2$$

$$\mathbf{w}^{(i,l+1)} = \mathbf{w}^{(i,l)} + \gamma^{(i,l)} \mathbf{p}^{(i,l)},$$

where gamma is the step length, J is the Jacobian matrix and P is a weighting matrix,

$$\mathbf{r}^{(i,l-1)} \equiv \mathbf{r}^{(i)}(\mathbf{w}^{(i,l-1)}, \boldsymbol{\mu}), \qquad \mathbf{J}^{(i,l-1)} \equiv \frac{\delta \mathbf{r}^{(i)}}{\delta \mathbf{w}}(\mathbf{w}^{(i,l-1)}, \boldsymbol{\mu})$$

$$\mathbf{P} = \begin{bmatrix} \mathbf{I} & 0 \\ 0 & \epsilon^2 \mathbf{I} \end{bmatrix}.$$

Regularization Approach

As a first (crude) attempt, I solved a regularized least squares problem instead of the true weighted least squares problem:

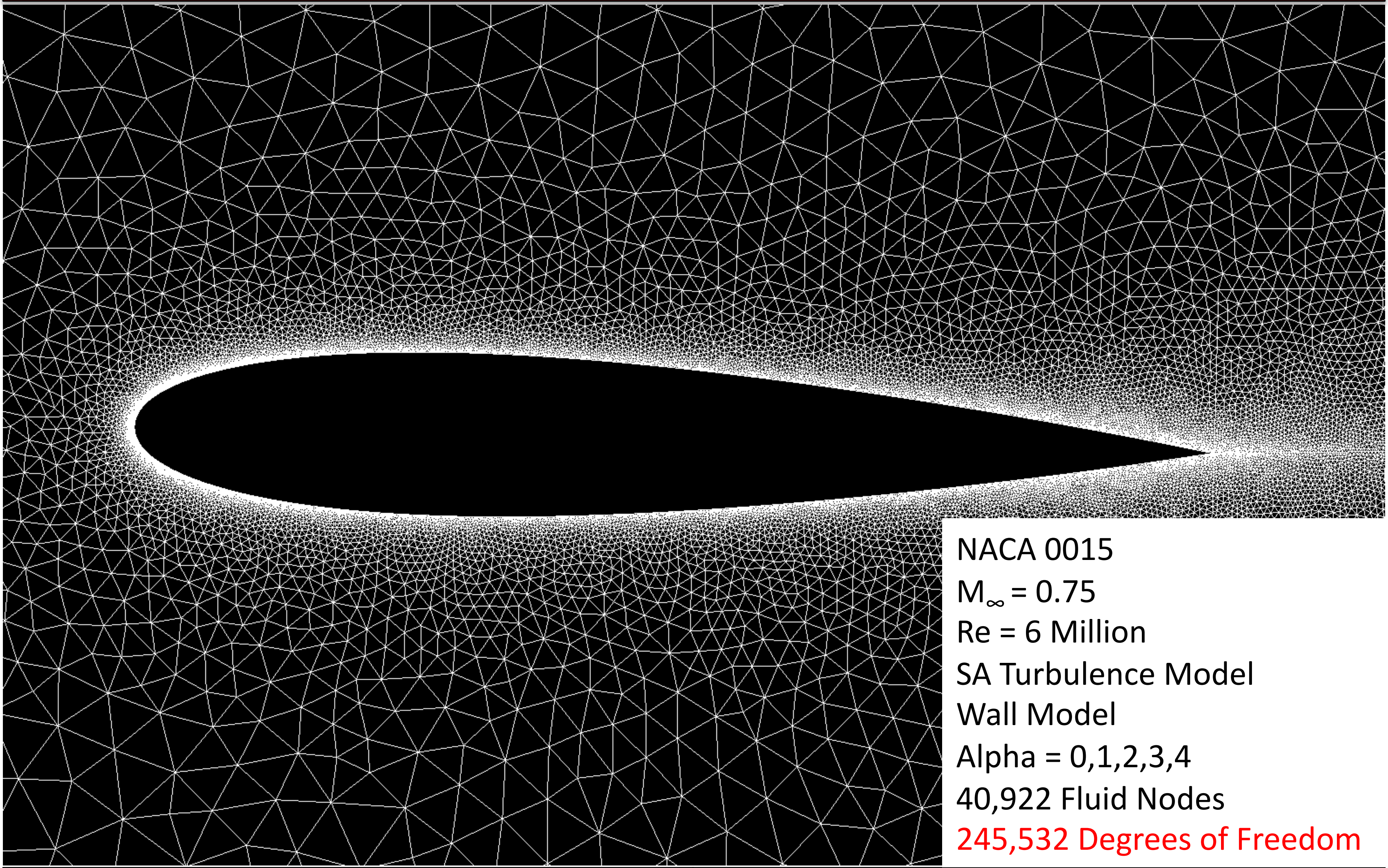
$$\min_{\Delta \mathbf{w}^{(i)} \in \mathbb{R}^n} \left(\left\| \mathbf{J}^{(i,l-1)} \mathbf{V} \Delta \mathbf{w}_k^{(i,l)} + \mathbf{r}^{(i,l-1)} \right\|_2^2 + \left\| \left(\mathbf{w}^{(i,l-1)} + \mathbf{V} \Delta \mathbf{w}_k^{(i,l)} \right) - \tilde{\mathbf{w}} \right\|_{\mathbf{Q}}^2 \right)$$

where $\tilde{\mathbf{w}}$ is an approximation to the true state (at least near the boundary) and the matrix Q defines a semi-norm.

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 \\ 0 & \epsilon^2 \mathbf{I} \end{bmatrix}.$$

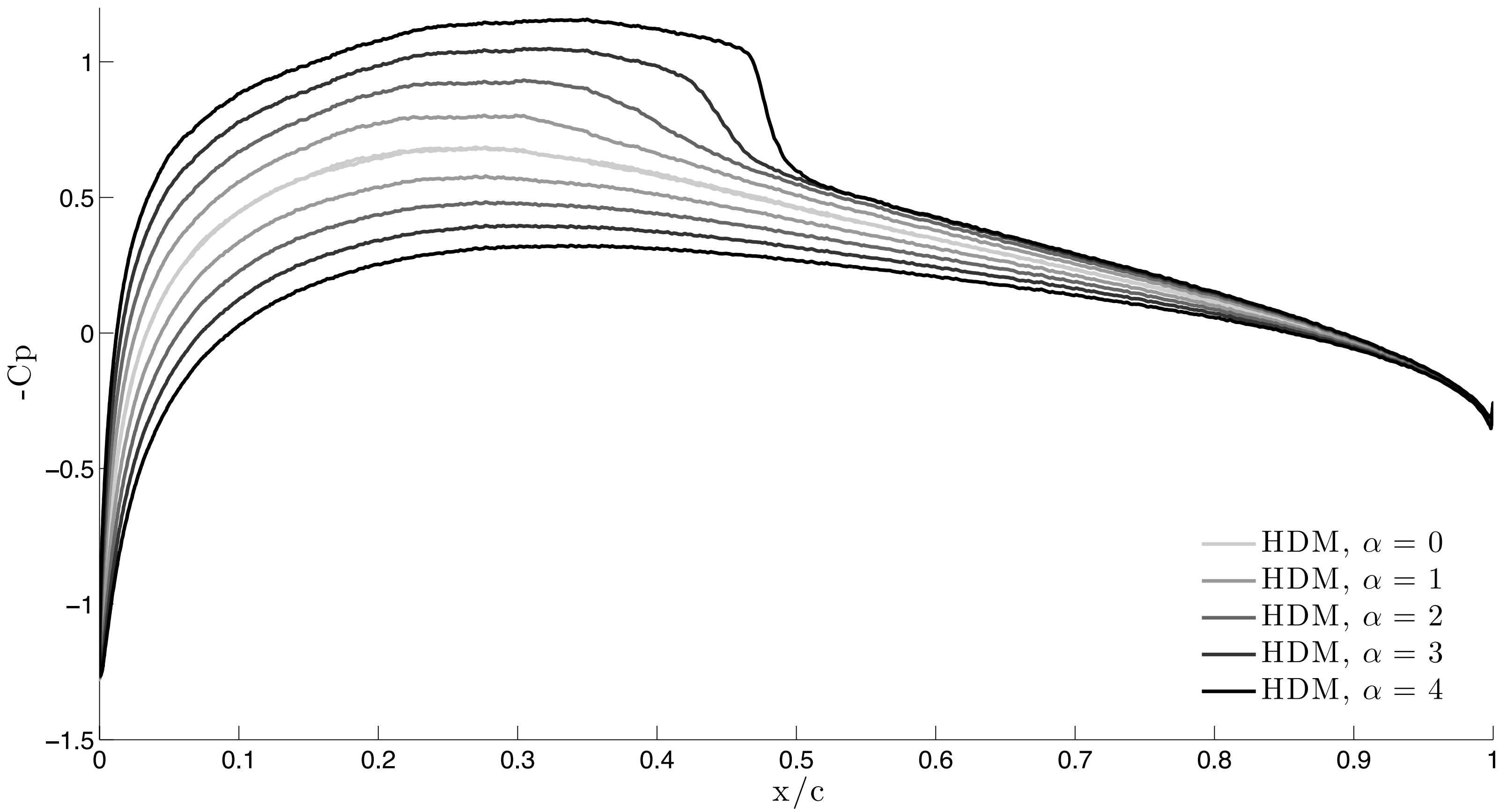
(Note: Q masks all interior nodes, so the regularization term is only active for the boundary nodes)

High-Dimensional Model (HDM)

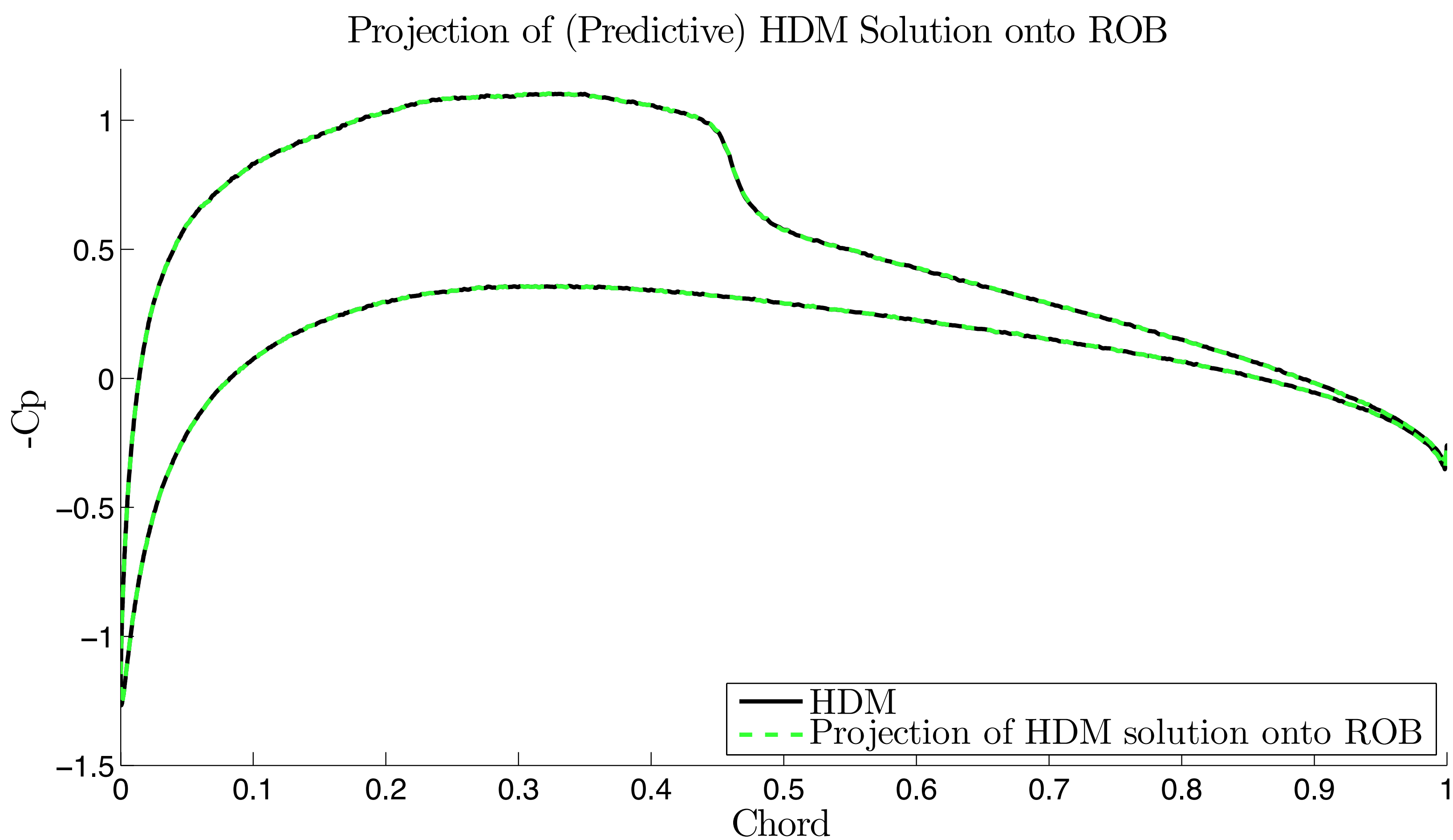


NACA 0015
$M_\infty = 0.75$
Re = 6 Million
SA Turbulence Model
Wall Model
Alpha = 0,1,2,3,4
40,922 Fluid Nodes
245,532 Degrees of Freedom

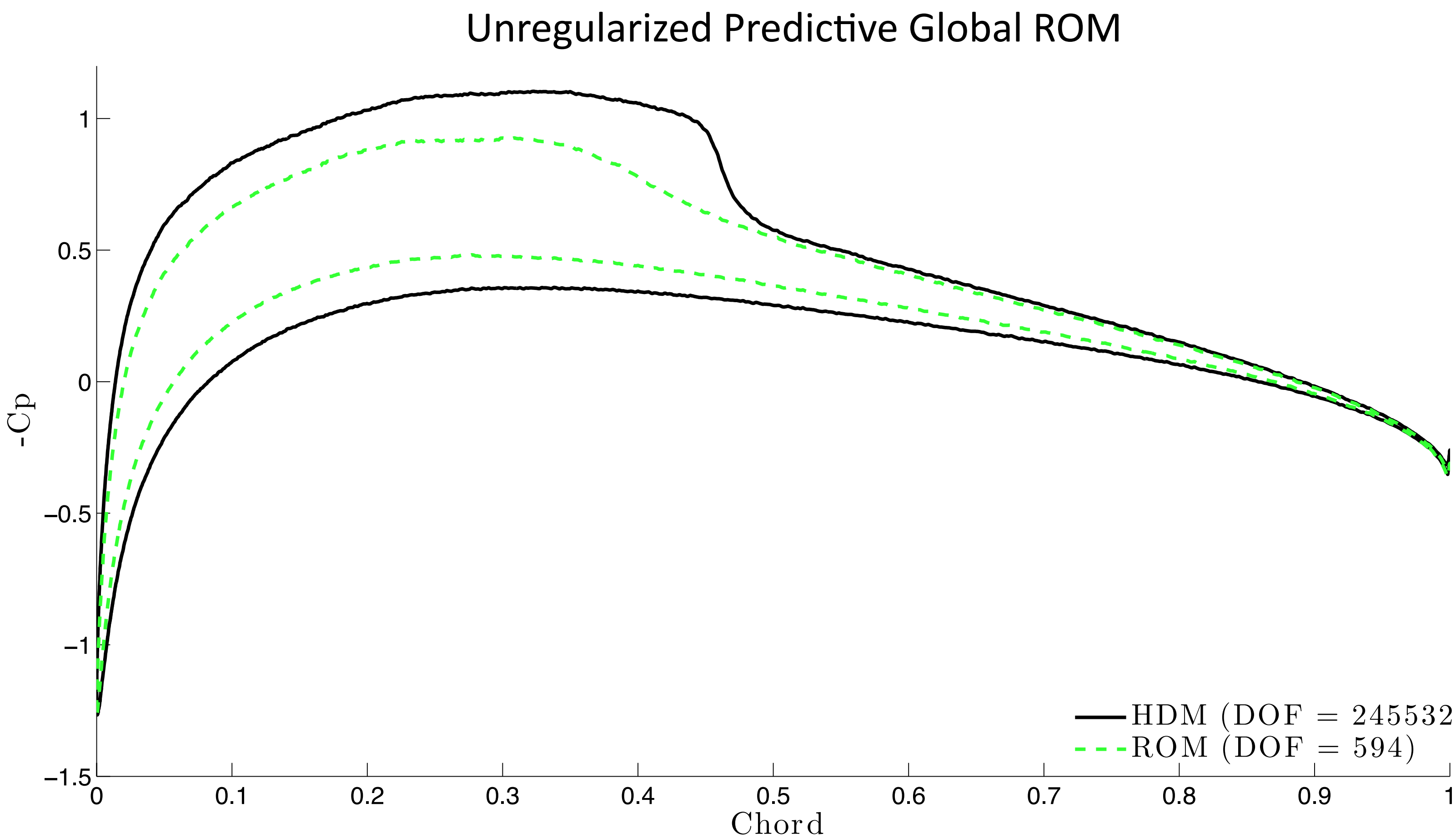
HDM Training Simulations



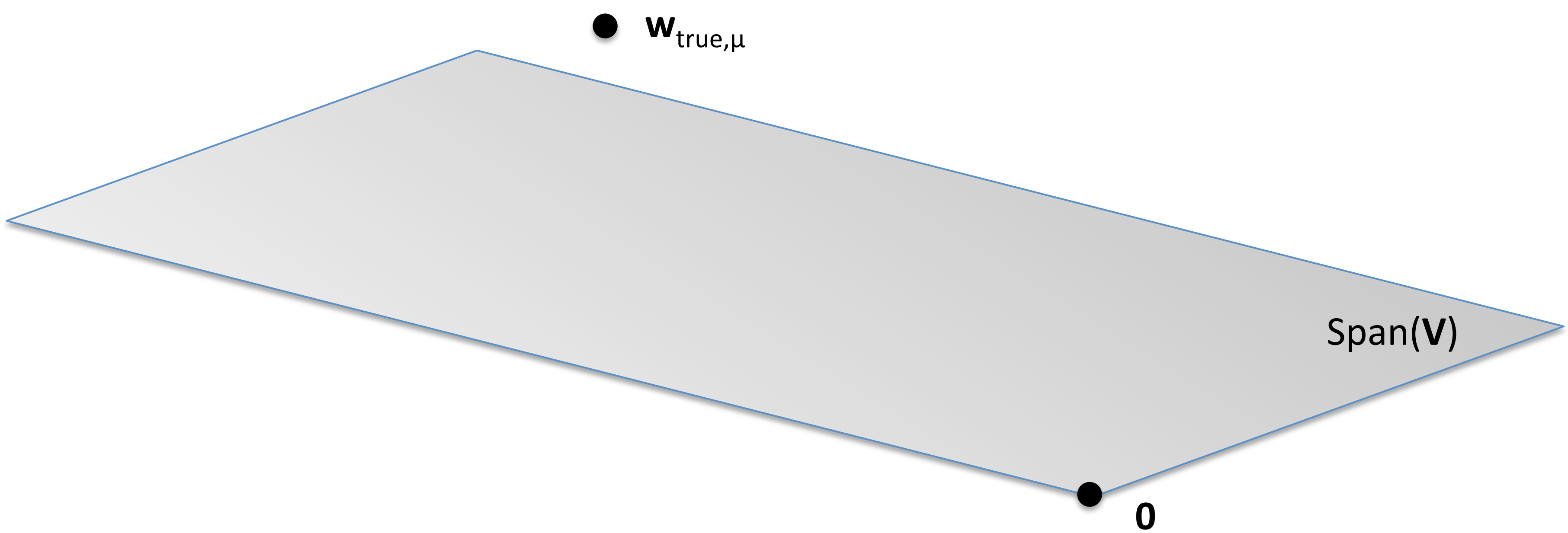
Predictive Operating Point (α=3.5)



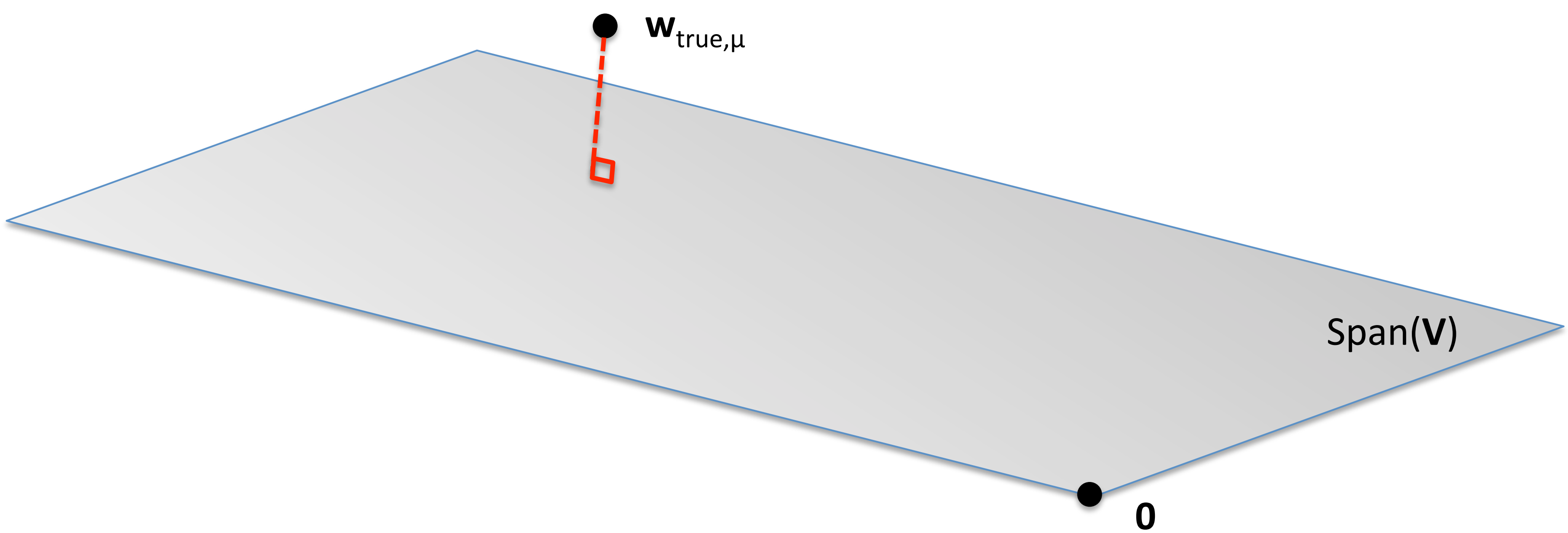
PG ROM, α=3.5 (Predictive)



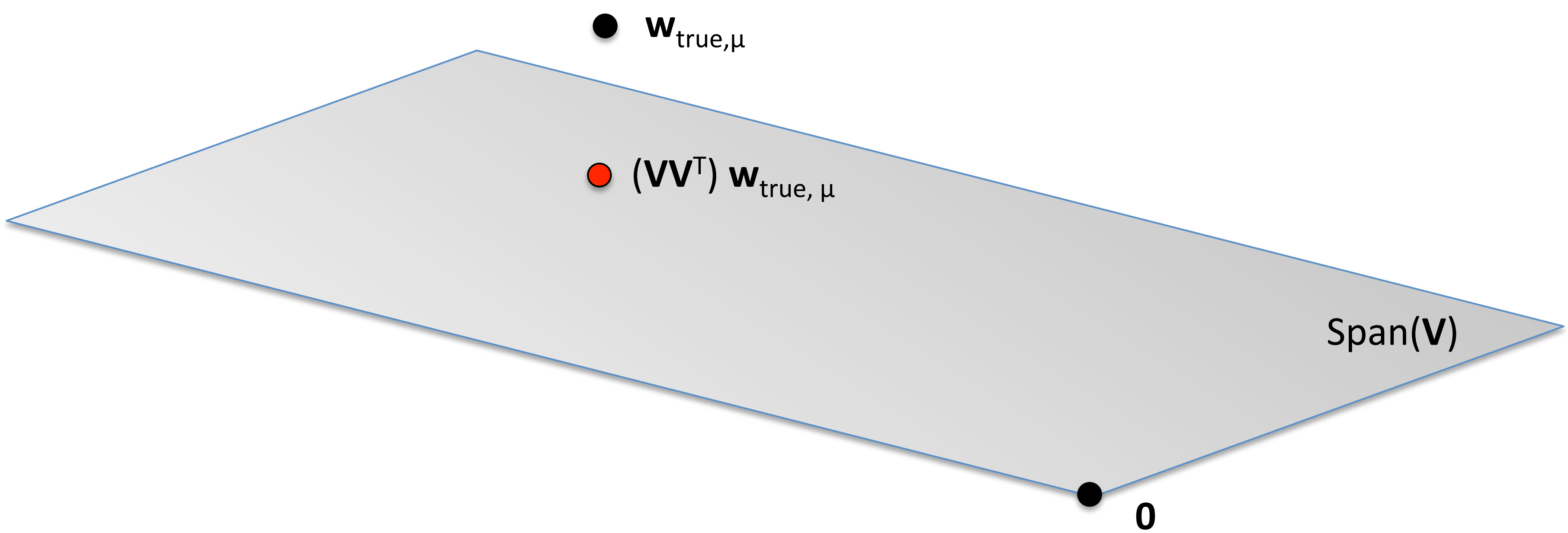
Predictive Operating Point (α=3.5)



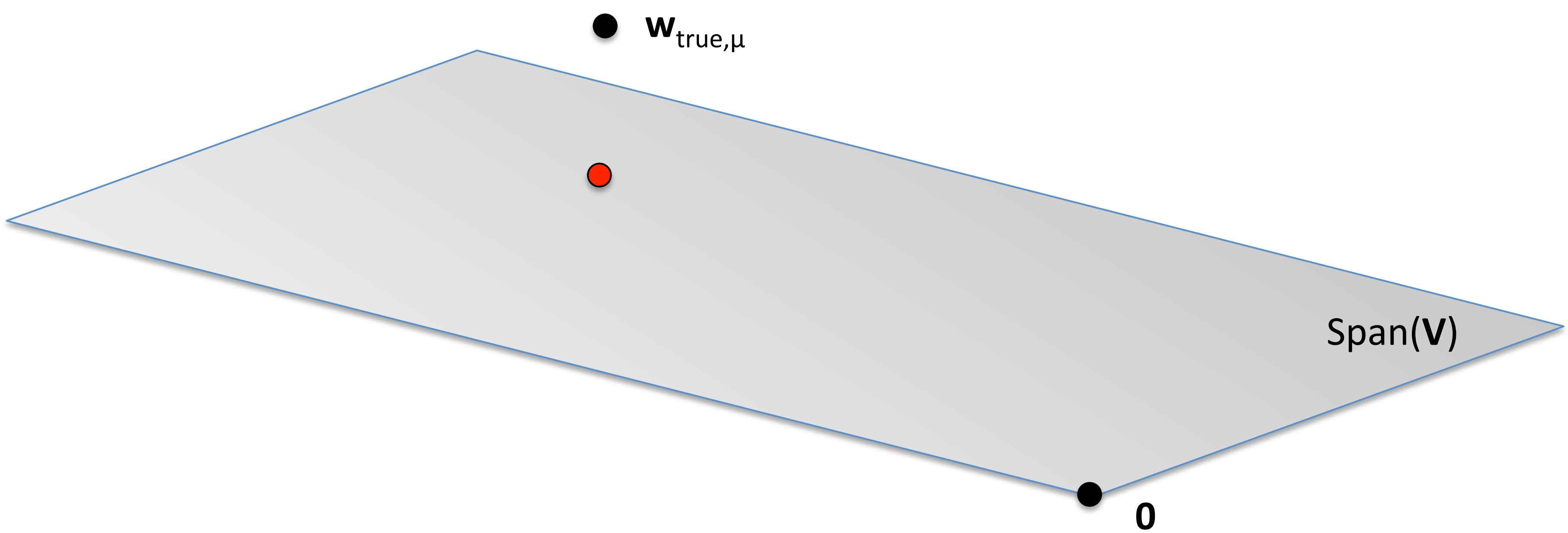
Predictive Operating Point (α=3.5)



Predictive Operating Point (α=3.5)

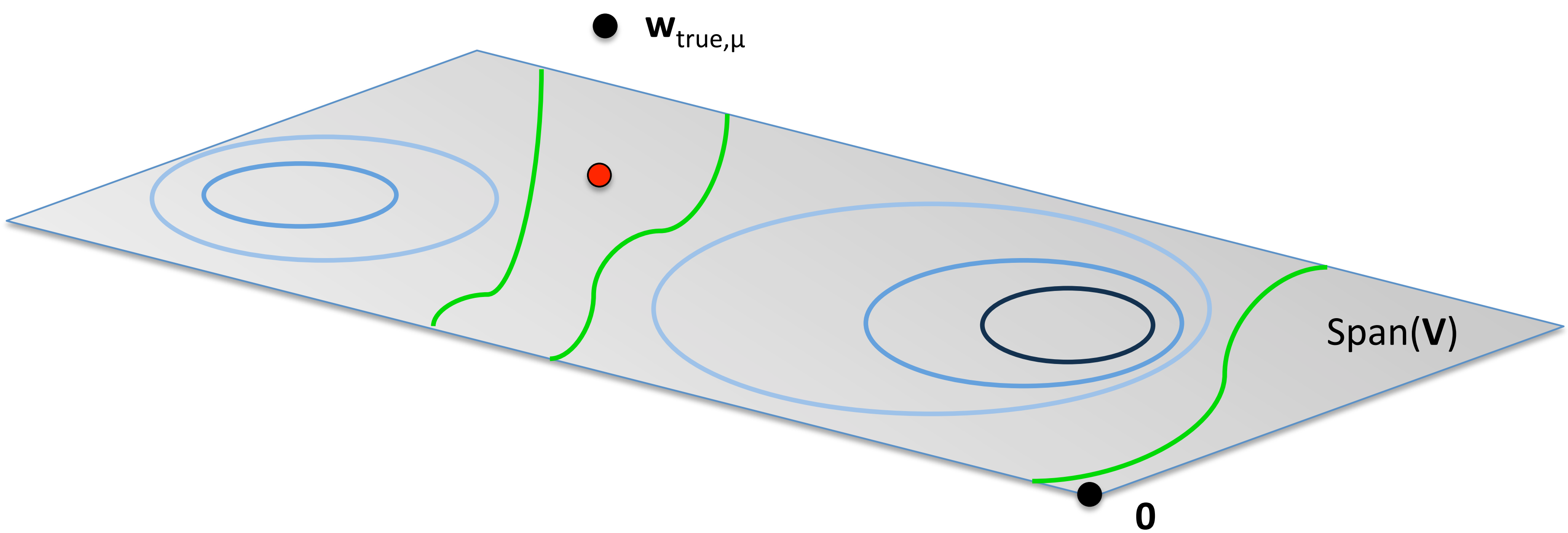


Predictive Operating Point (α=3.5)



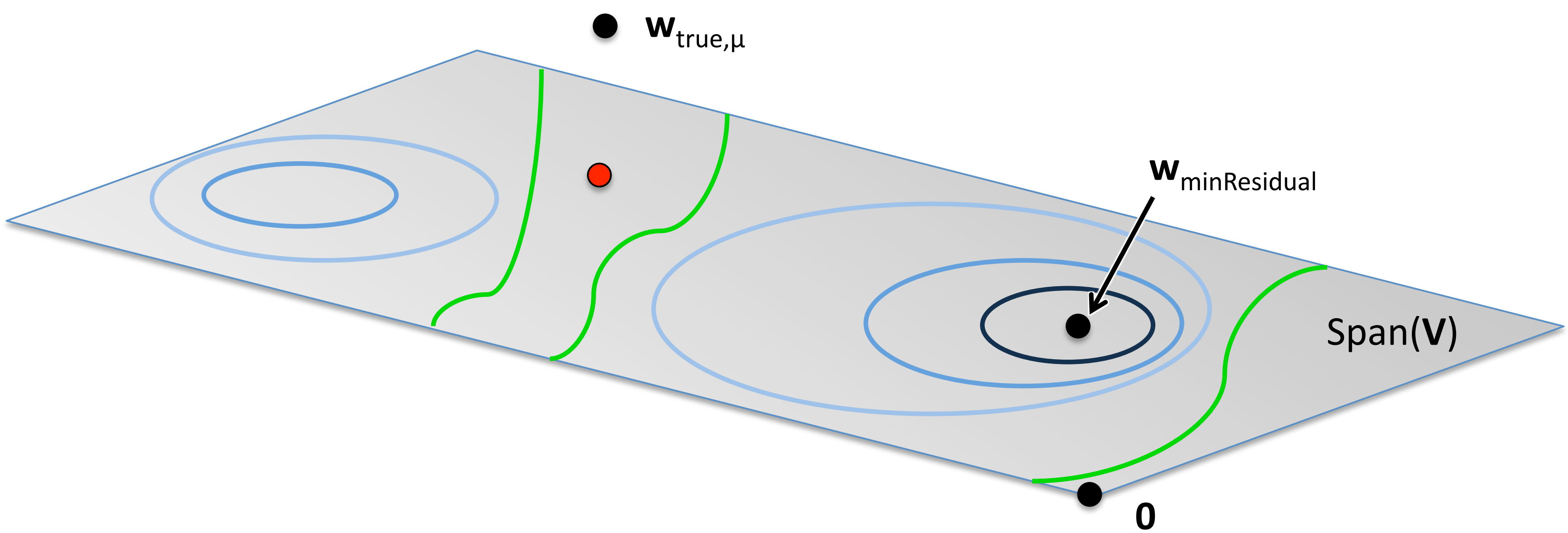
Predictive Operating Point (α=3.5)

Contours of unweighted residual: $\left\|\mathbf{r}^{(i)}\right\|_2^2$



Predictive Operating Point (α=3.5)

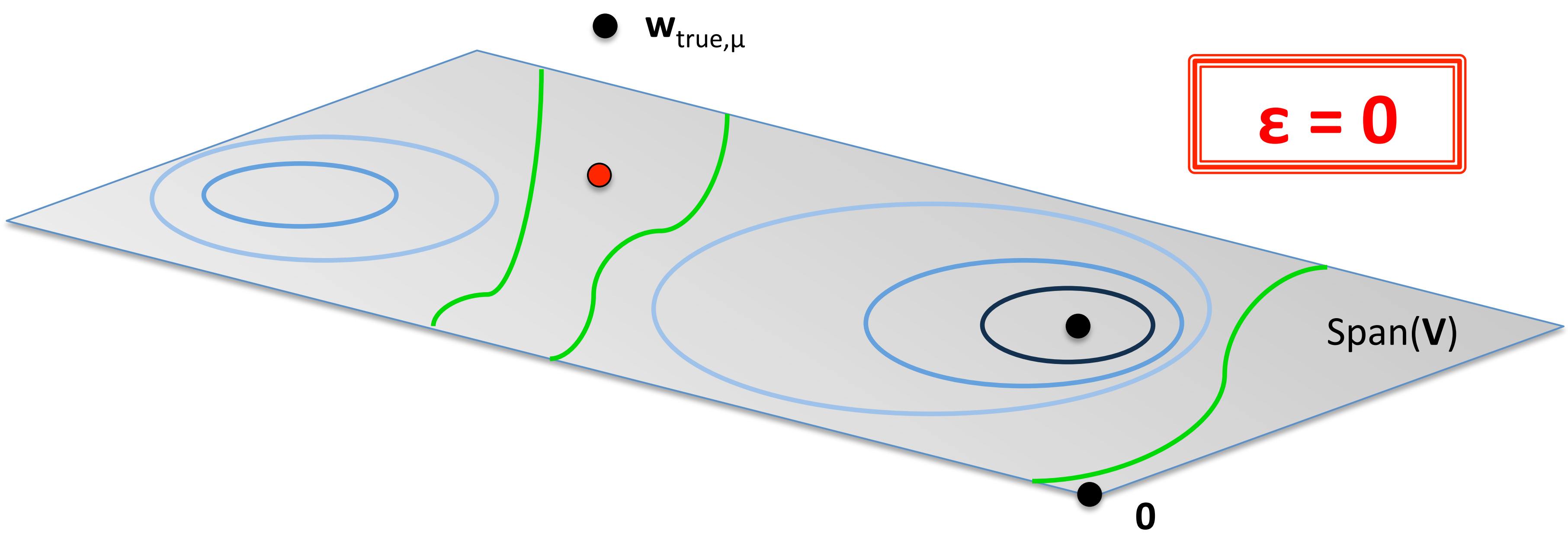
Contours of raw (HDM) residual: $\left\|\mathbf{r}^{(i)}\right\|_2^2$



Predictive Operating Point (α=3.5)

Contours of regularized residual: $\left\|\mathbf{r}^{(i)}\right\|_2^2 + \left\|\mathbf{w}^{(i)} - \tilde{\mathbf{w}}\right\|_{\mathbf{Q}}^2$

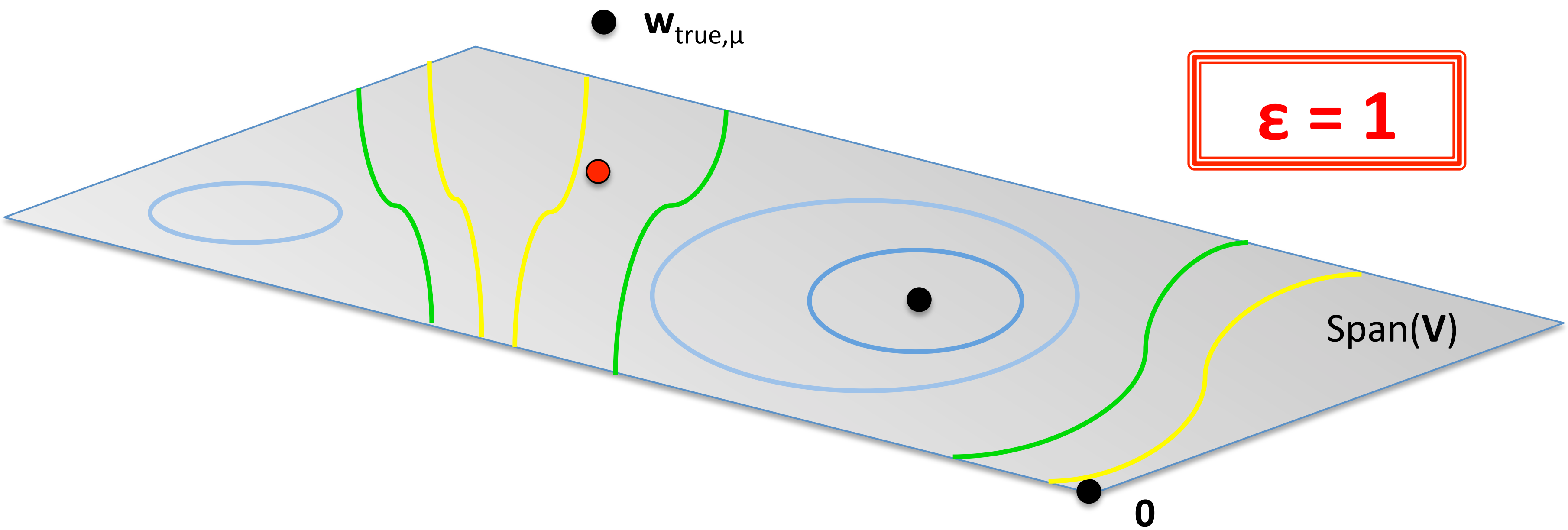
where $\mathbf{Q} = \begin{bmatrix} 0 & 0 \\ 0 & \epsilon^2\mathbf{I} \end{bmatrix}.$



Predictive Operating Point (α=3.5)

Contours of regularized residual: $\left\|\mathbf{r}^{(i)}\right\|_2^2 + \left\|\mathbf{w}^{(i)} - \tilde{\mathbf{w}}\right\|_{\mathbf{Q}}^2$

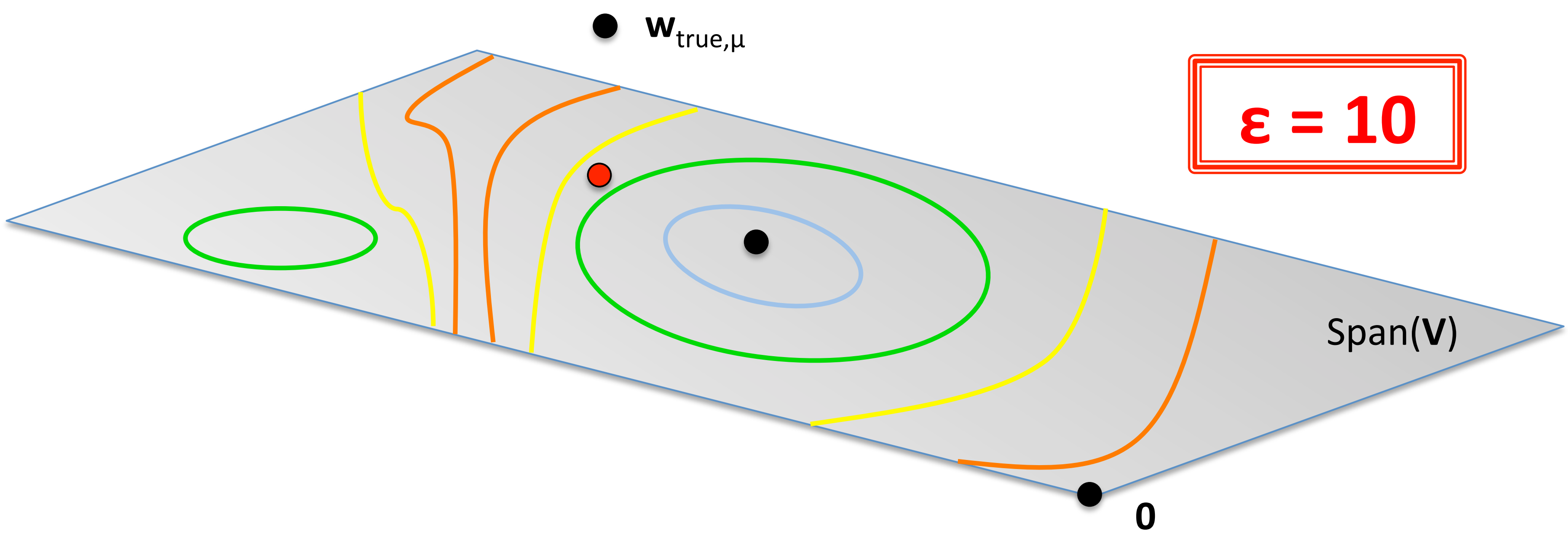
where $\mathbf{Q} = \begin{bmatrix} 0 & 0 \\ 0 & \epsilon^2\mathbf{I} \end{bmatrix}.$



Predictive Operating Point (α=3.5)

Contours of regularized residual: $$\left\|\mathbf{r}^{(i)}\right\|_2^2 + \left\|\mathbf{w}^{(i)} - \tilde{\mathbf{w}}\right\|_{\mathbf{Q}}^2$$

where $$\mathbf{Q} = \begin{bmatrix} 0 & 0 \\ 0 & \epsilon^2 \mathbf{I} \end{bmatrix}.$$

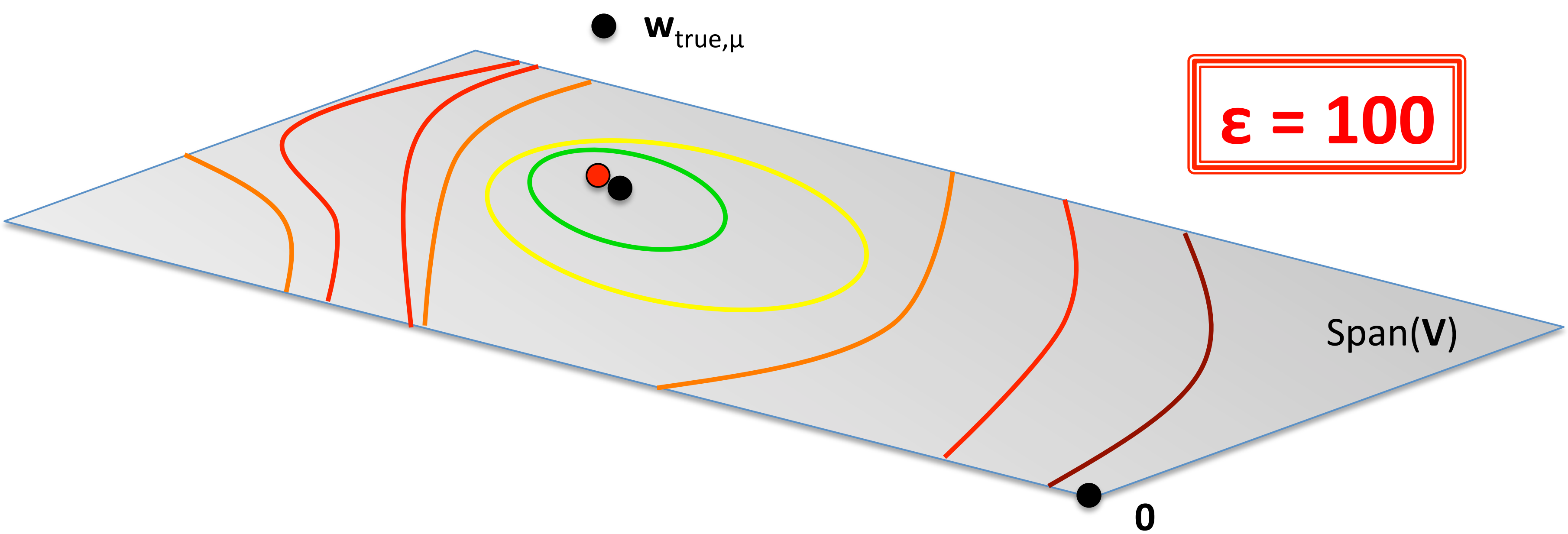


ε = 10

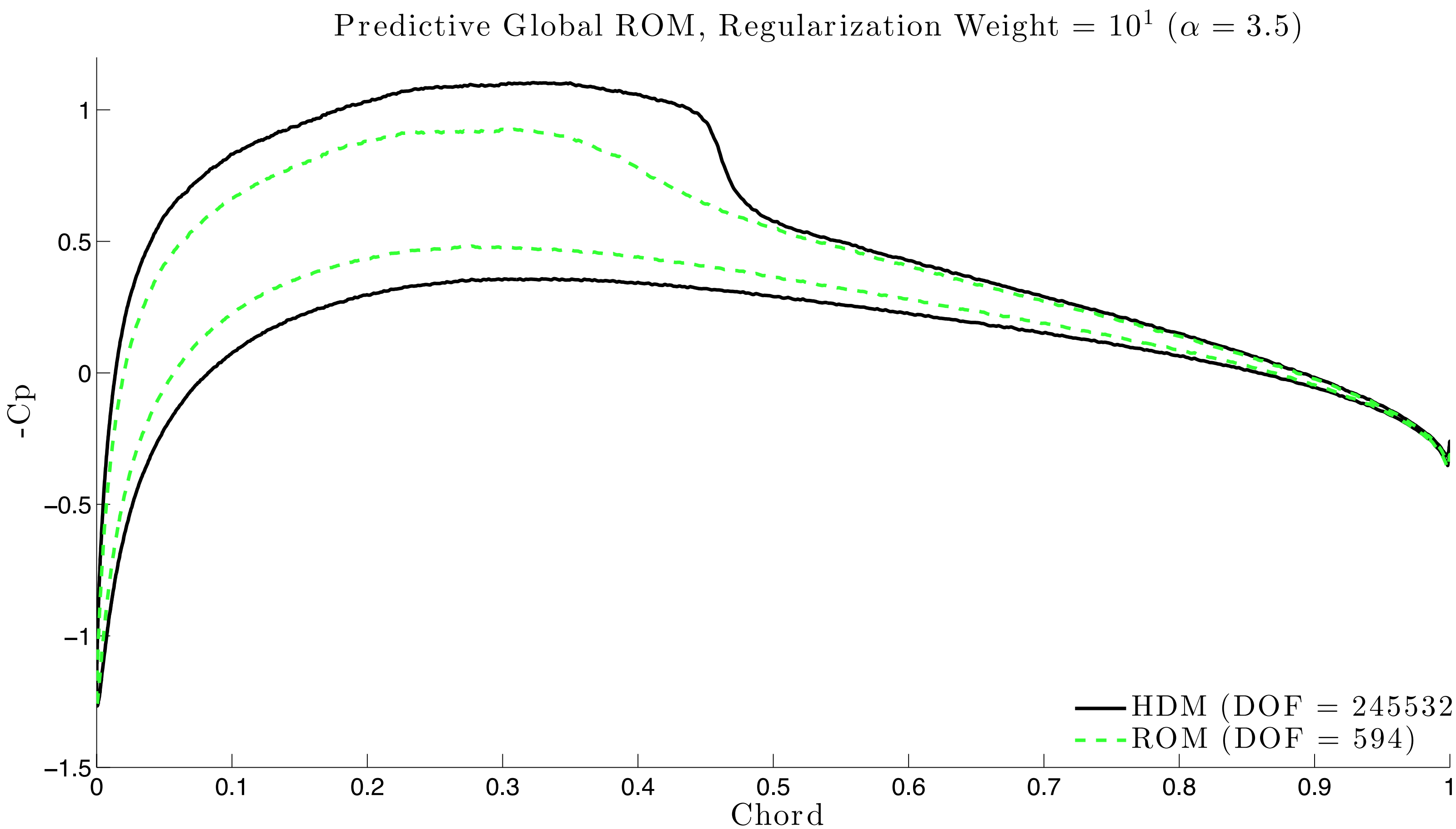
Predictive Operating Point (α=3.5)

Contours of regularized residual: $\left\| \mathbf{r}^{(i)} \right\|_2^2 + \left\| \mathbf{w}^{(i)} - \tilde{\mathbf{w}} \right\|_{\mathbf{Q}}^2$

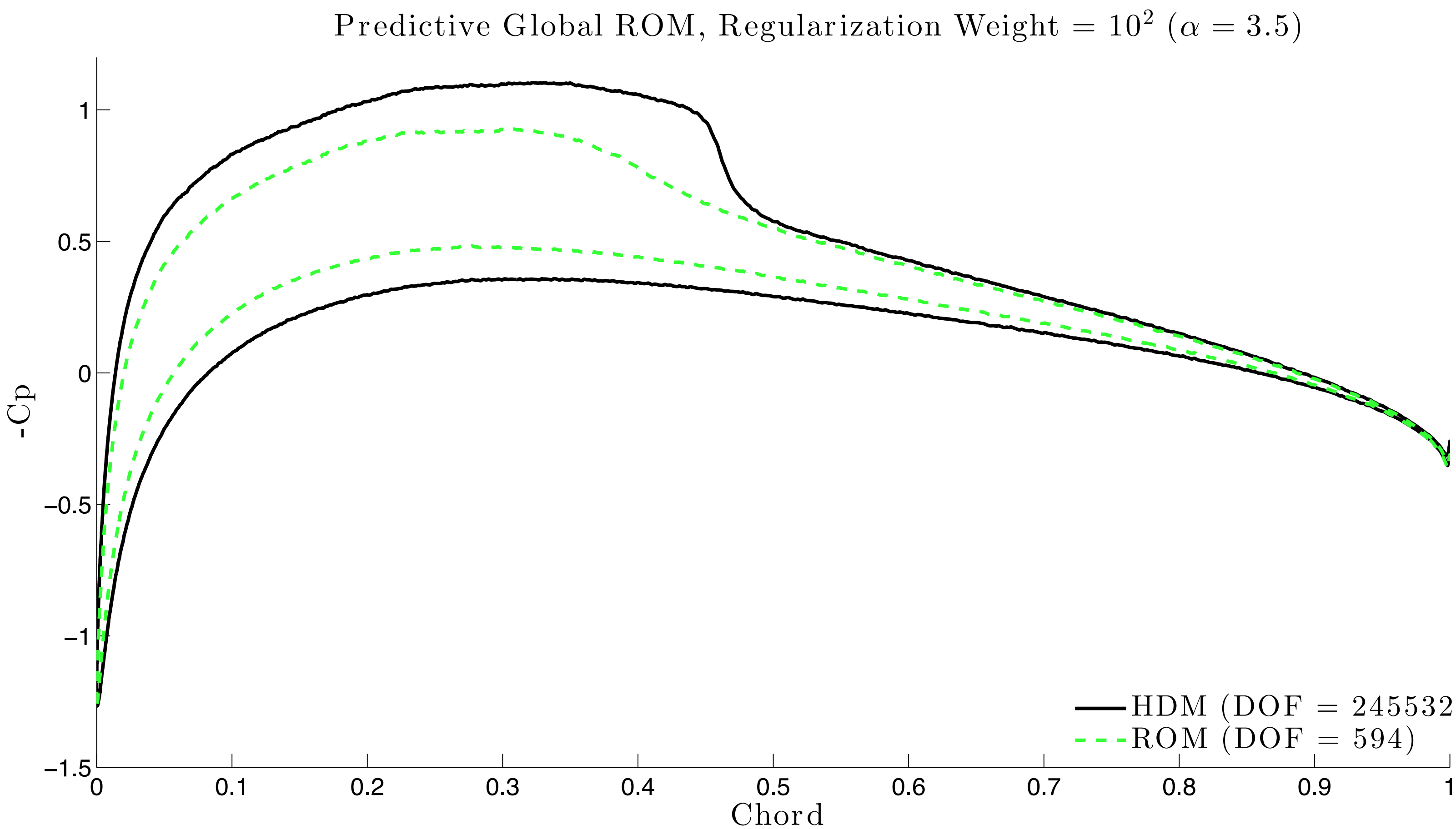
where $\mathbf{Q} = \begin{bmatrix} 0 & 0 \\ 0 & \epsilon^2 \mathbf{I} \end{bmatrix}.$



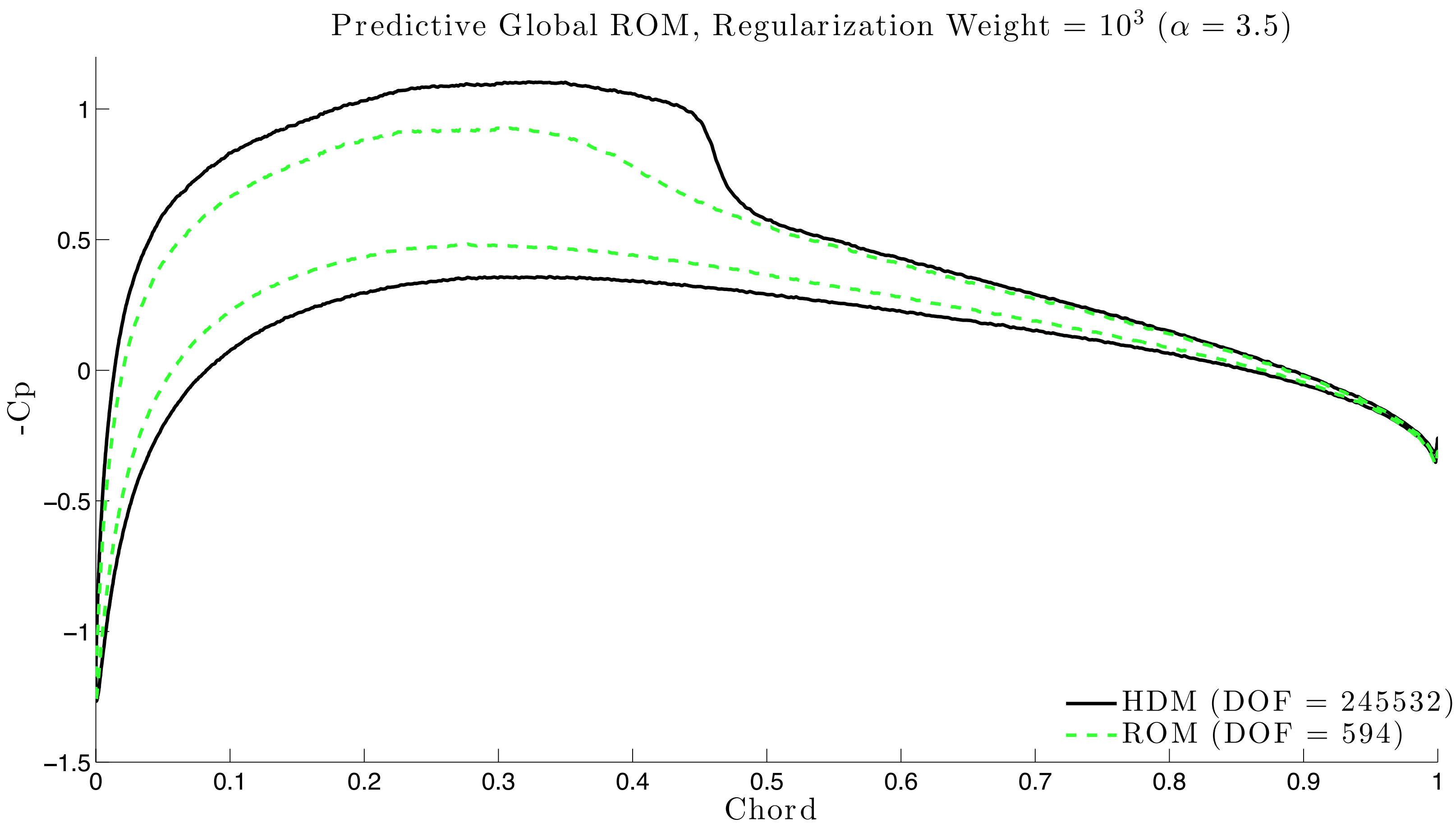
PG ROM, α=3.5 (Predictive), ε = 10



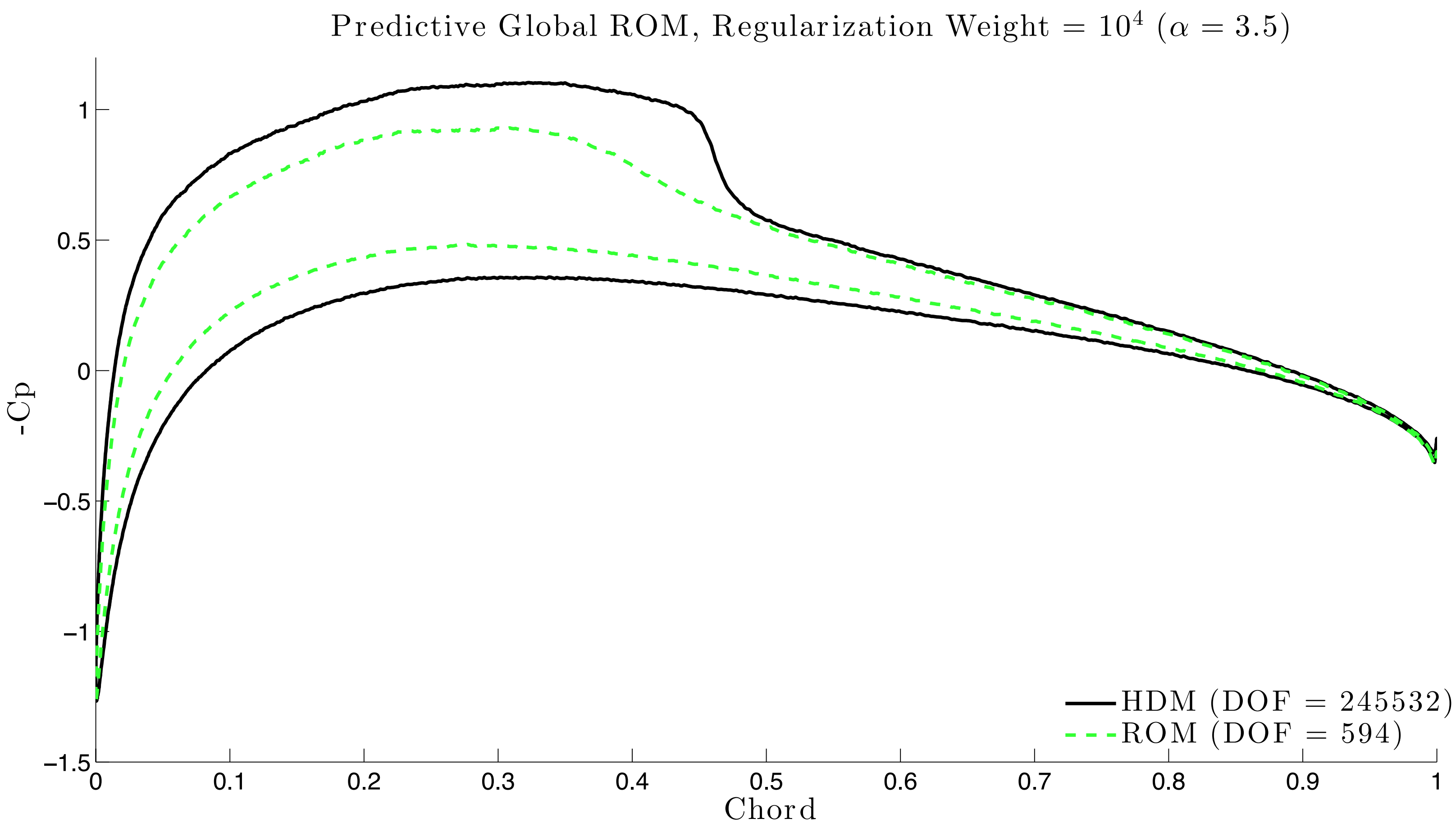
PG ROM, α=3.5 (Predictive), ε = 10^2



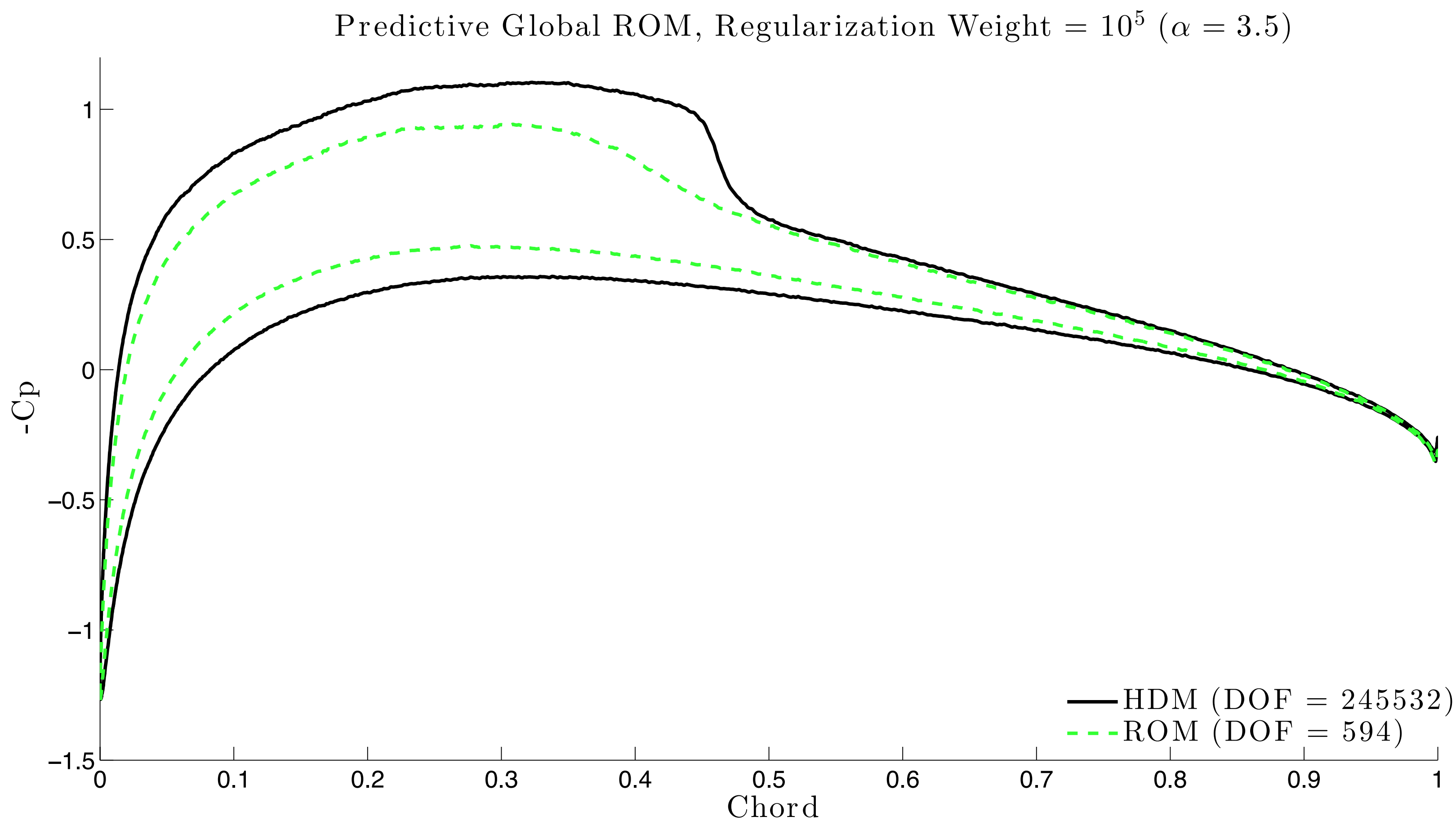
PG ROM, α=3.5 (Predictive), $\varepsilon = 10^3$



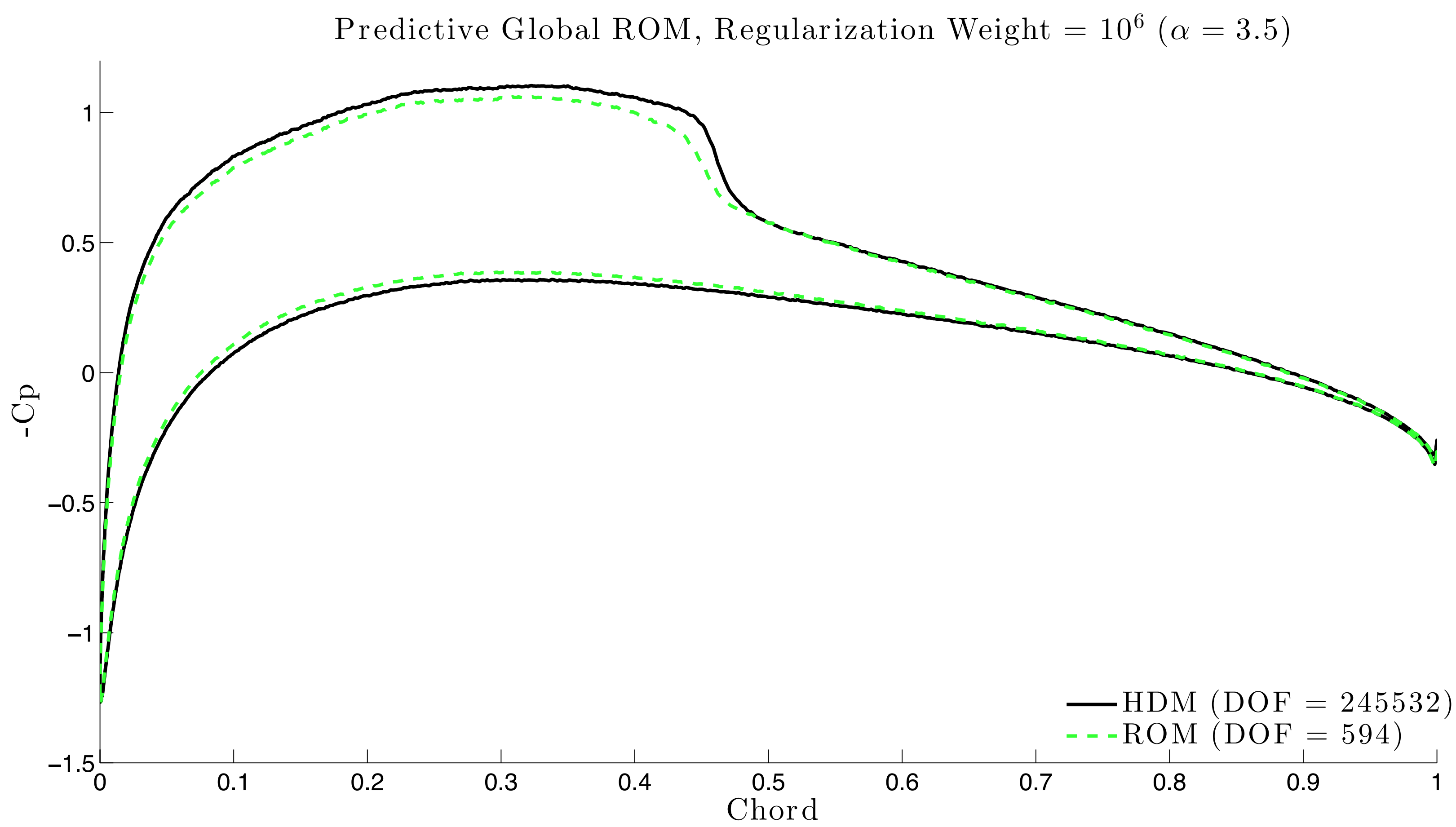
PG ROM, α=3.5 (Predictive), ε = 10^4



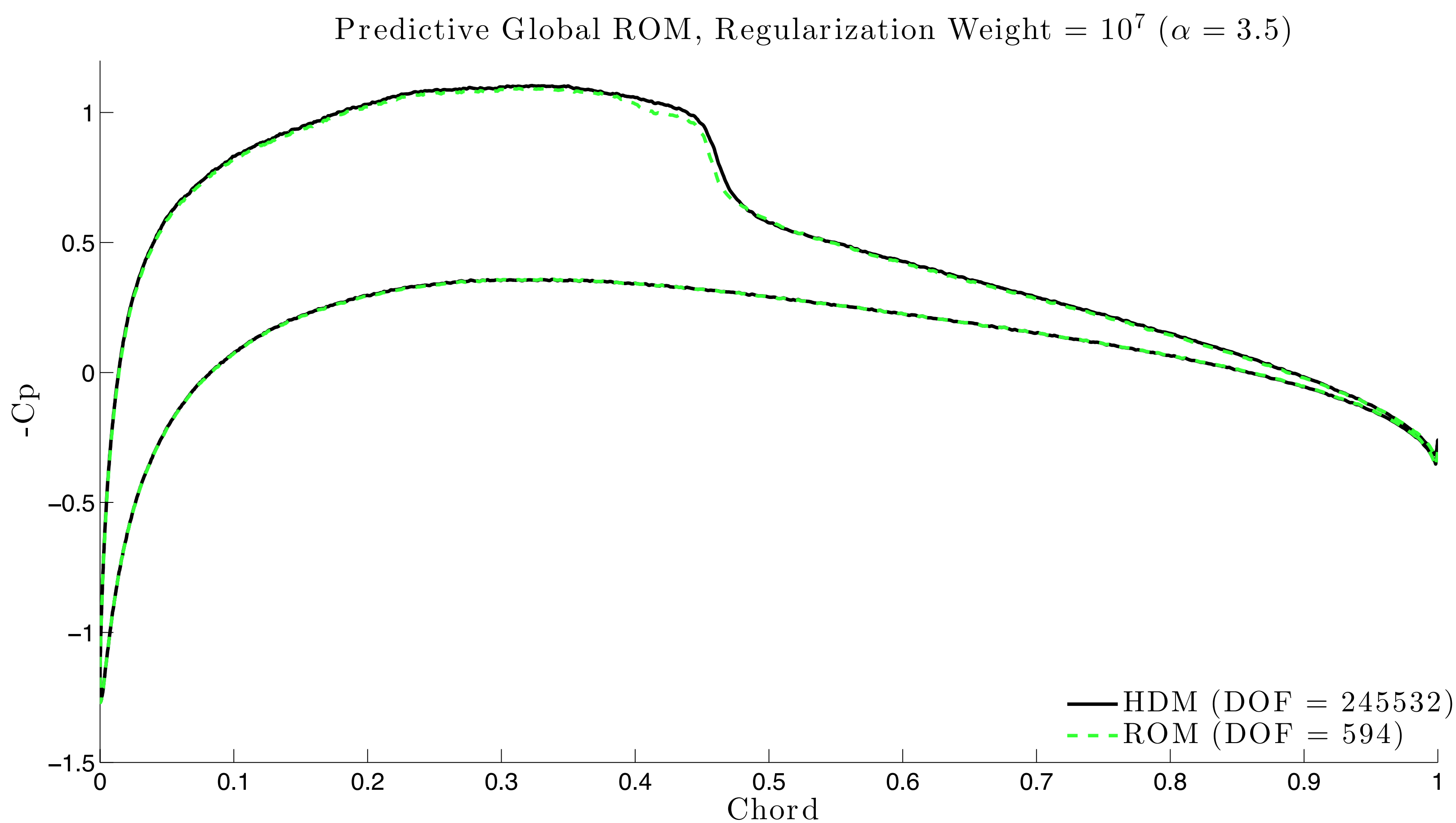
PG ROM, α=3.5 (Predictive), ε = 10^5



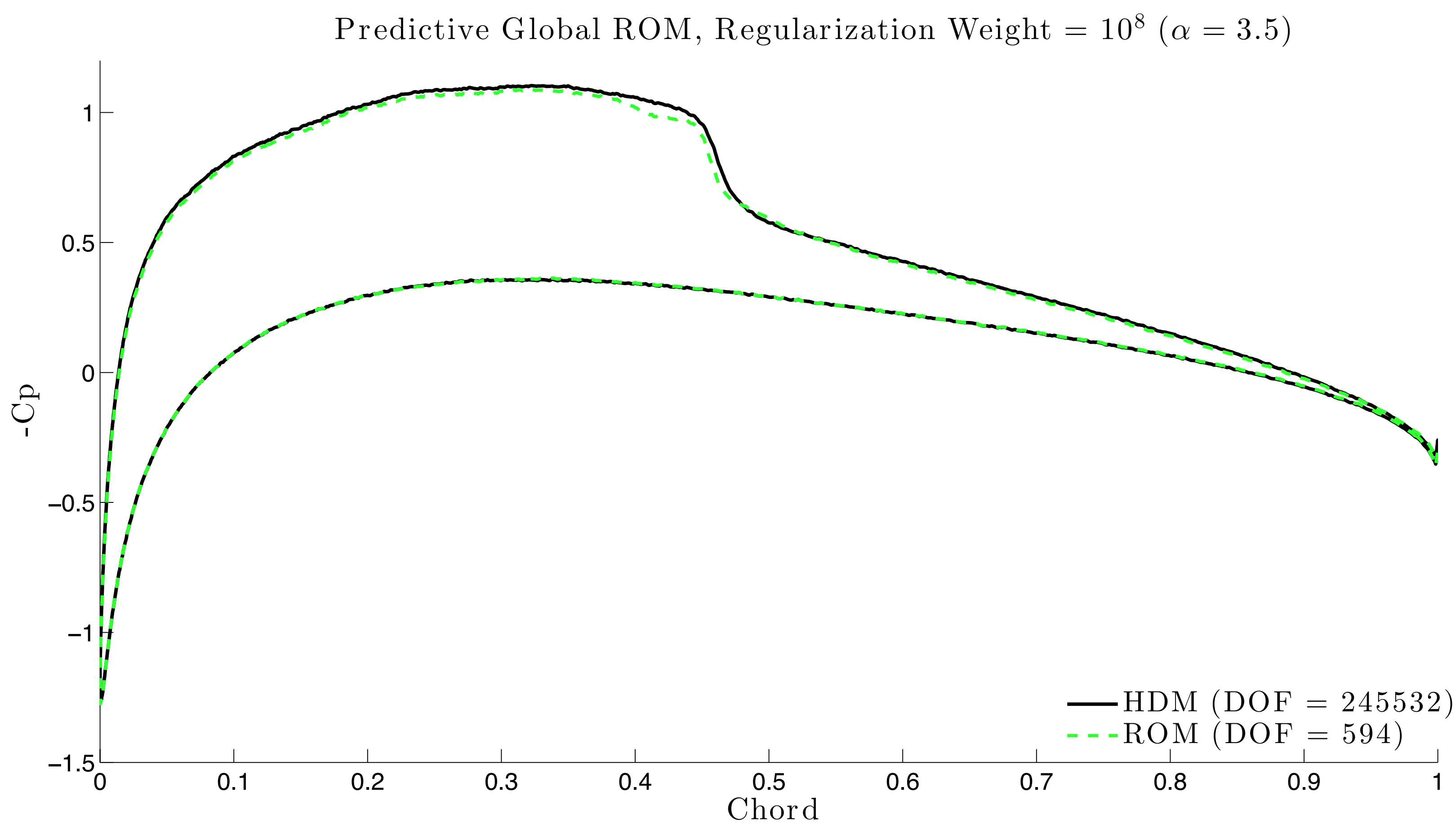
PG ROM, α=3.5 (Predictive), ε = 10^6



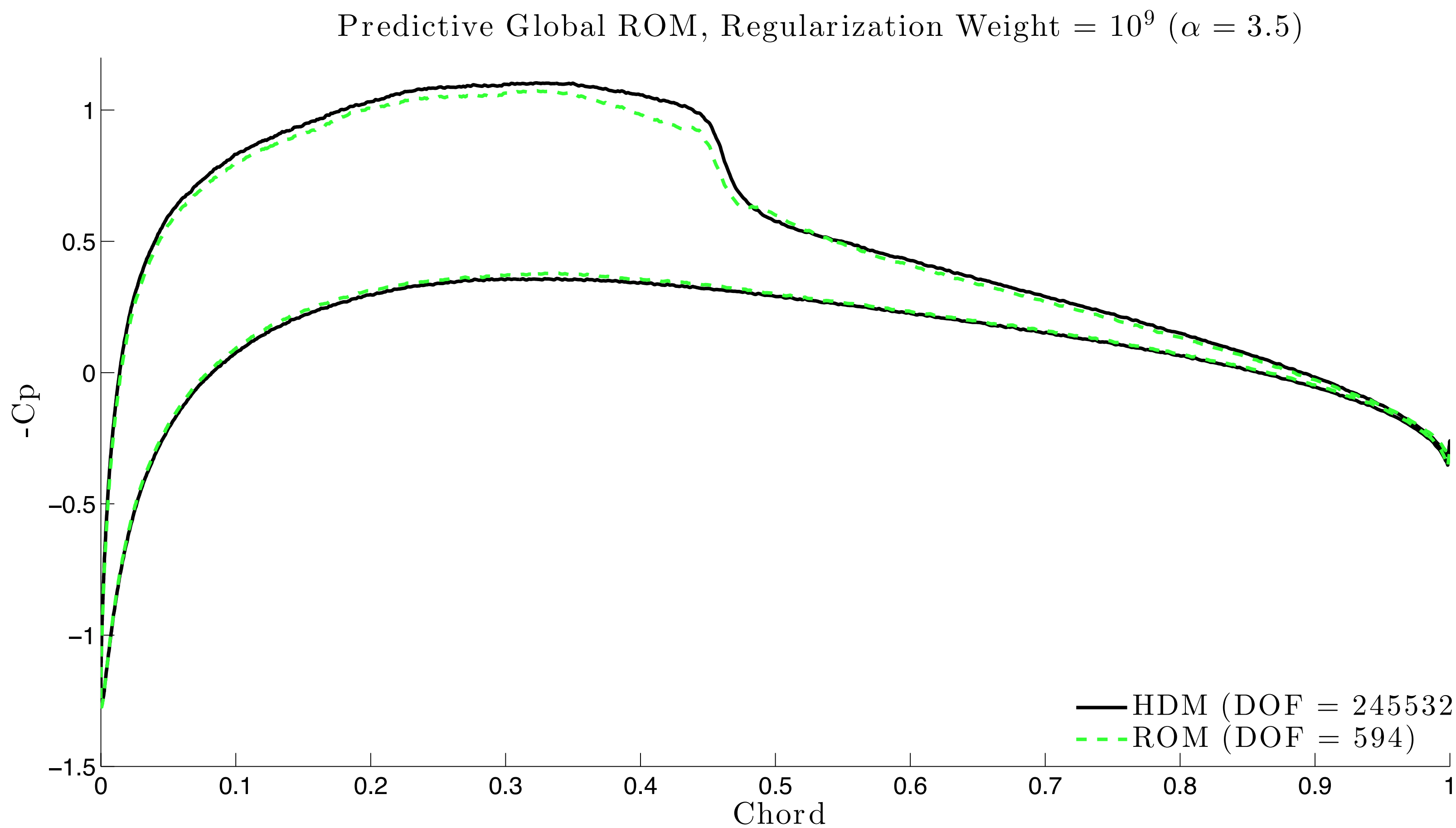
PG ROM, α=3.5 (Predictive), ε = 10^7



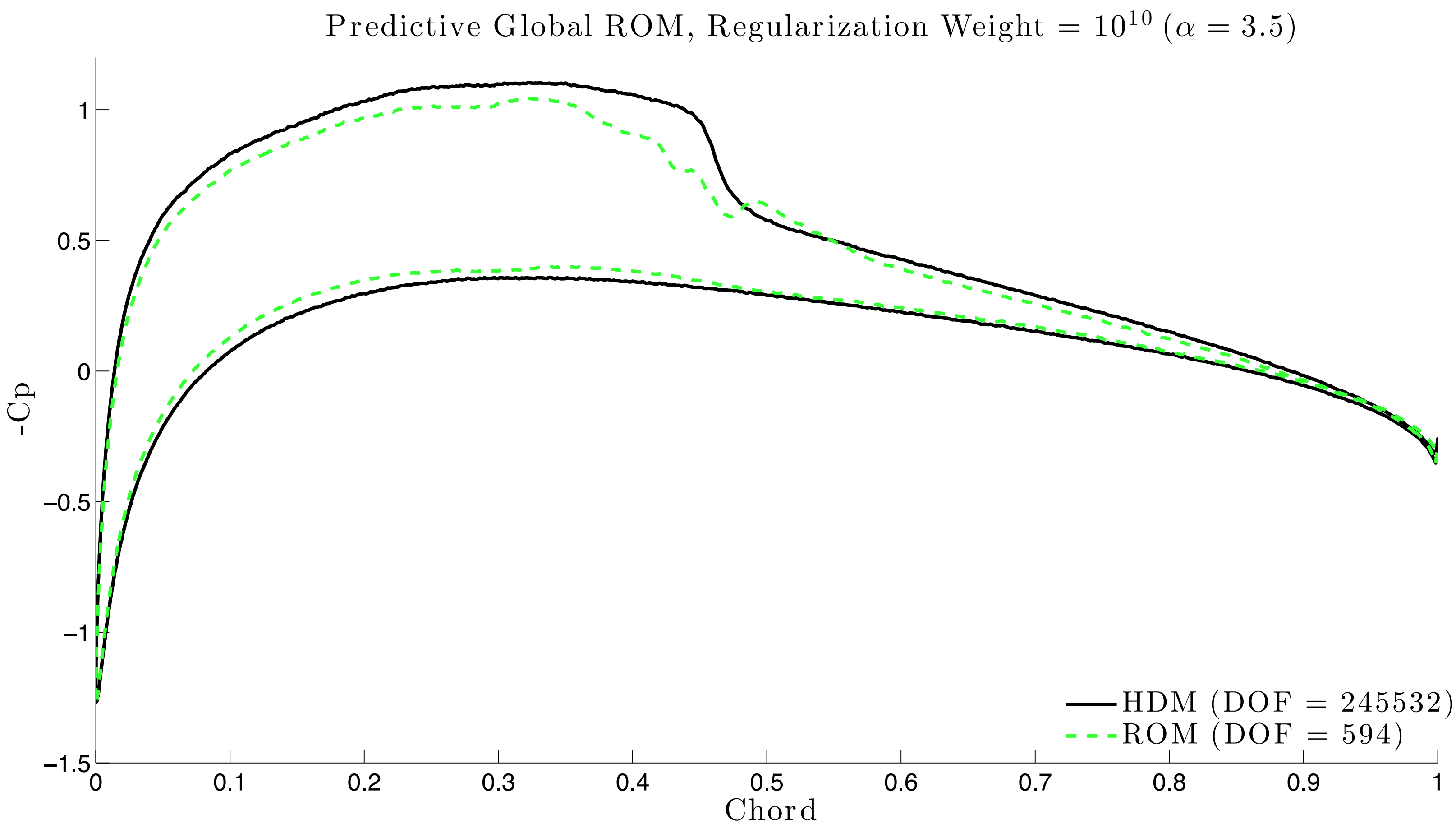
PG ROM, α=3.5 (Predictive), ε = 10^8



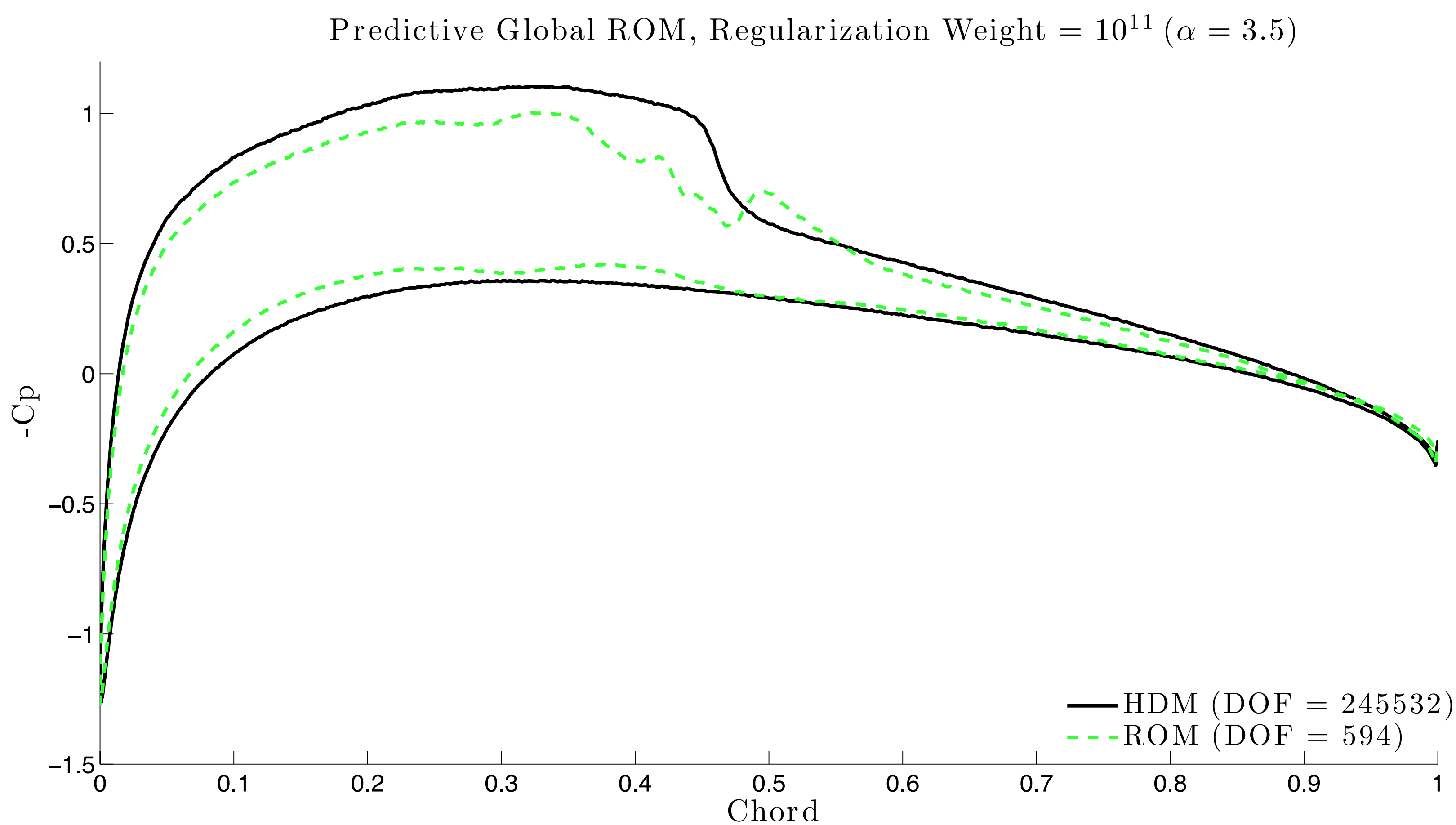
PG ROM, α=3.5 (Predictive), $\varepsilon = 10^9$



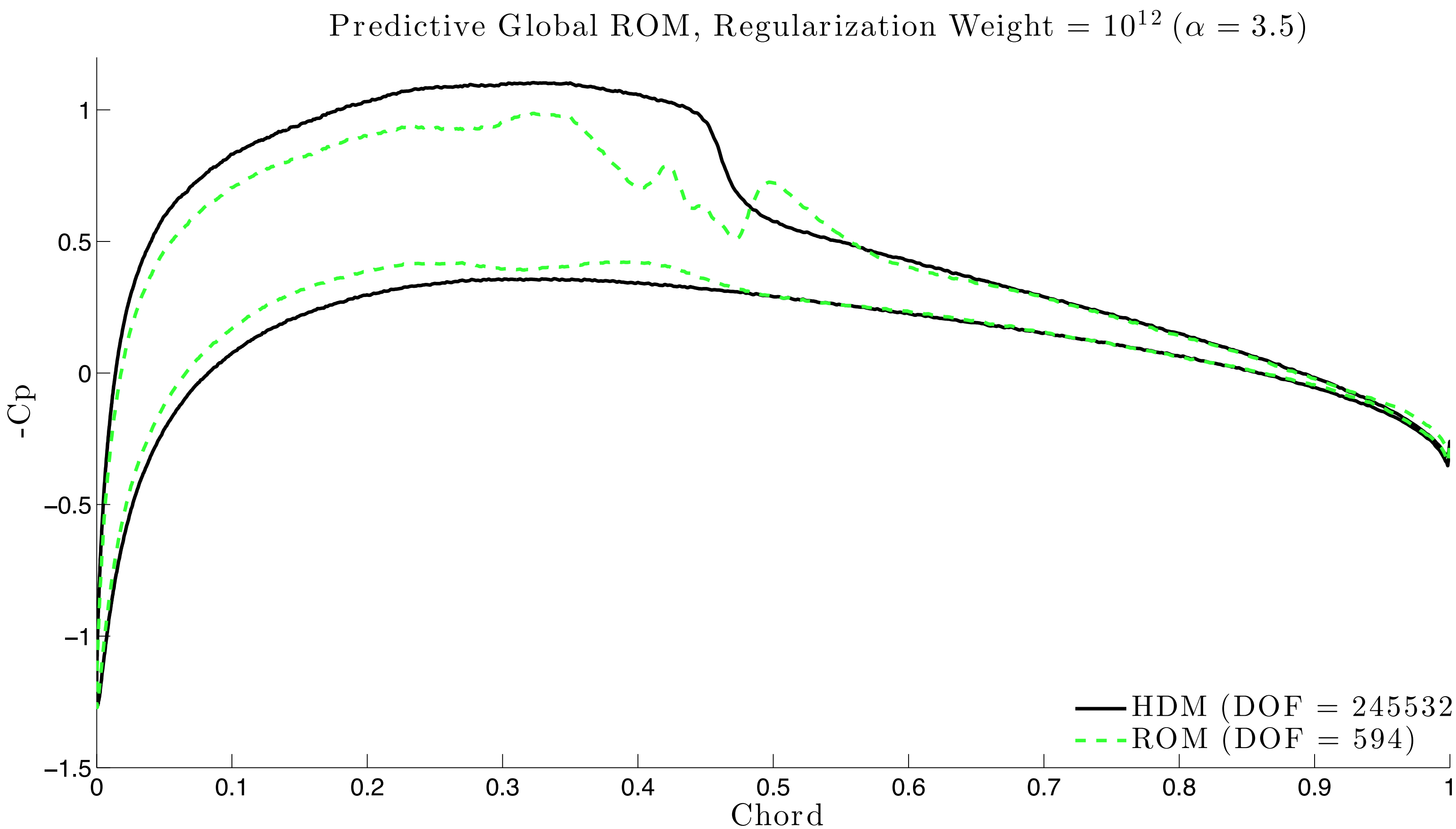
PG ROM, α=3.5 (Predictive), ε = 10^{10}

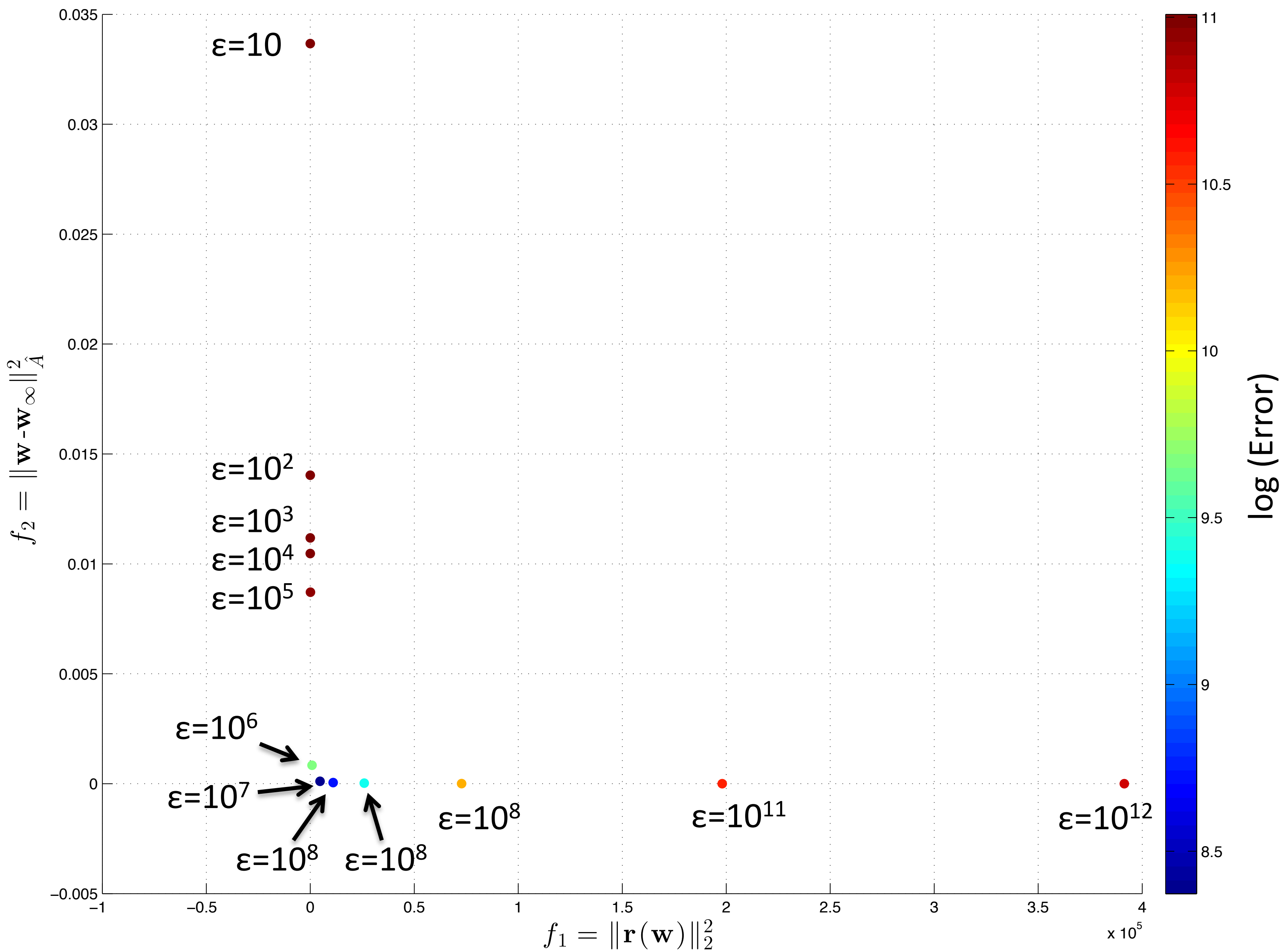


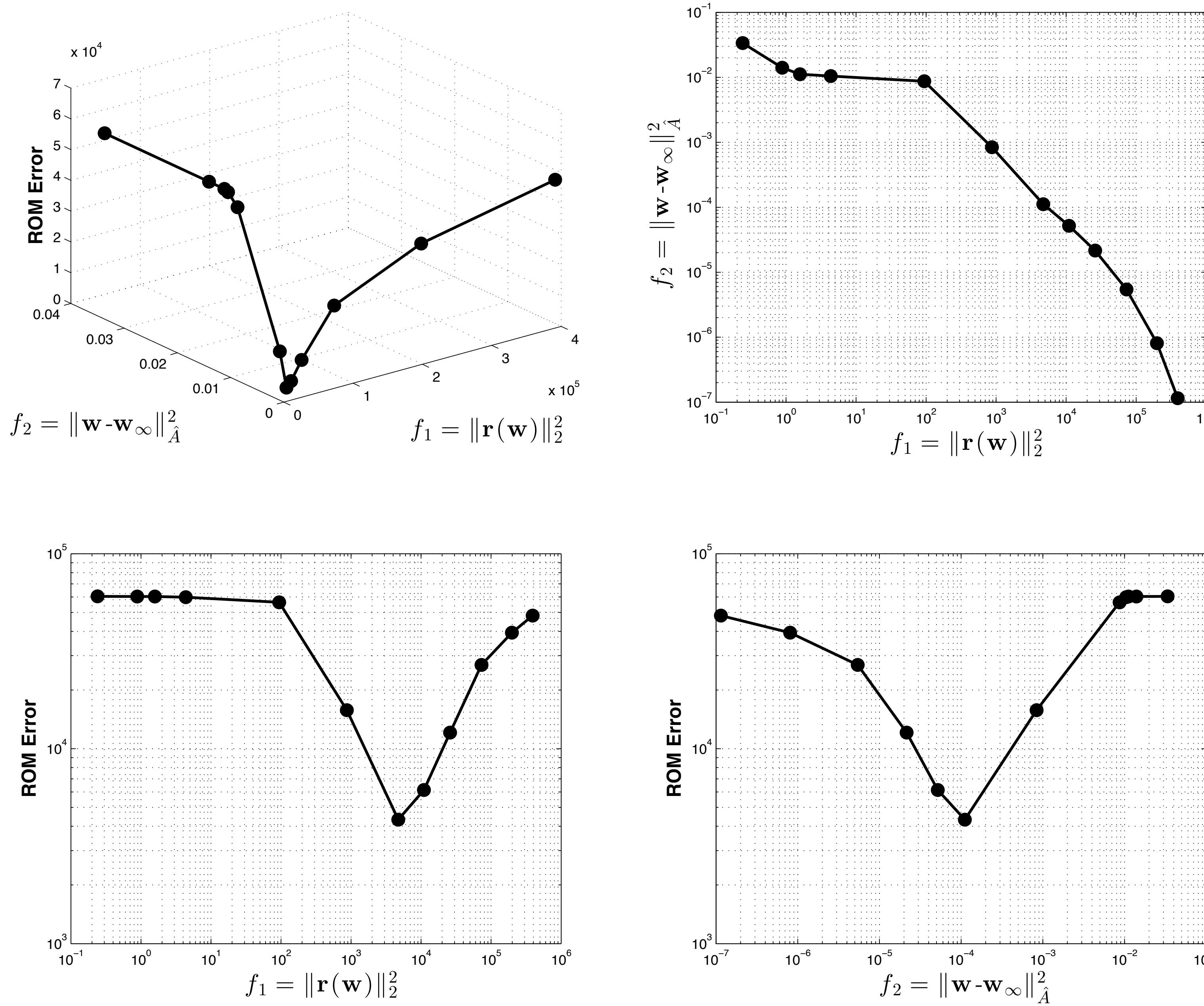
PG ROM, α=3.5 (Predictive), ε = 10^{11}



PG ROM, α=3.5 (Predictive), ε = 10^{12}




ε=10
ε=10²
ε=10³
ε=10⁴
ε=10⁵
ε=10⁶
ε=10⁷
ε=10⁸
ε=10⁸
ε=10⁸
ε=10¹¹
ε=10¹²
$f_2 = \|\mathbf{w}-\mathbf{w}_\infty\|^2_{\hat{A}}$
$f_1 = \|\mathbf{r}(\mathbf{w})\|^2_2$
x 10^5
log (Error)


ROM Error
x 10^4
0
1
2
3
4
5
6
7
0.04
0.03
0.02
0.01
0
1
2
3
4
x 10^5
$f_2 = \|\mathbf{w} - \mathbf{w}_\infty\|_{\hat{A}}^2$
$f_1 = \|\mathbf{r}(\mathbf{w})\|_2^2$
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
$f_2 = \|\mathbf{w} - \mathbf{w}_\infty\|_{\hat{A}}^2$
$f_1 = \|\mathbf{r}(\mathbf{w})\|_2^2$
ROM Error
10^{5}
10^{4}
10^{3}
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
$f_1 = \|\mathbf{r}(\mathbf{w})\|_2^2$
ROM Error
10^{5}
10^{4}
10^{3}
10^{-7}
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
10^{-1}
$f_2 = \|\mathbf{w} - \mathbf{w}_\infty\|_{\hat{A}}^2$

Conclusions / Future Work

- The corner of this plot seems to correspond to low errors in the ROM results.
- In the next few weeks I'll implement a method to find this corner automatically and efficiently.
- I'm also planning to implement the weighted least squares formulation of the problem rather than the regularized least squares formulation. It will be interesting to see whether the weighted least squares formulation exhibits the same behavior as the regularized formulation as the weighting on the boundary is increased.

Supporting Slides

Regularizing Petrov-Galerkin

Standard least-squares:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_{\mathbf{P}}^2$$

$$\mathbf{x}^* = \left(\mathbf{A}^T\mathbf{P}\mathbf{A}\right)^{-1}\left(\mathbf{A}^T\mathbf{P}\mathbf{b}\right)$$

Least-squares with Tikhonov regularization (generalized form):

$$\min_{\mathbf{x}\in\mathbb{R}^n} \left(\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_{\mathbf{P}}^2 + \|\mathbf{x}-\tilde{\mathbf{x}}\|_{\mathbf{Q}}^2\right), \text{ where } \tilde{\mathbf{x}} \equiv \mathbf{E}(\mathbf{x}^*).$$

$$\mathbf{x}^* = \left(\mathbf{A}^T\mathbf{P}\mathbf{A}+\mathbf{Q}\right)^{-1}\left(\mathbf{A}^T\mathbf{P}\mathbf{b}+\mathbf{Q}\tilde{\mathbf{x}}\right)$$

Projection-Based Model Reduction

- **Residual minimization in the least squares sense**

$$w_r(t) = \underset{z}{\operatorname{argmin}} \, || \, r(w_0 + V_w z, t) \, ||_2$$

$$J(w) : \text{Jacobian of } r(. \, ,t) \text{ at } w$$

- **Solution by Newton's method (Gauss-Newton iterations)**

$$(J(w^{(m)})V_w)^T(J(w^{(m)})V_w) \, z^{(m)} = -(J(w^{(m)})V_w)^T \, r(w^{(m)}, t)$$

Projection-Based Model Reduction