Suppose we are given a partial differential equation (PDE):

$$\frac{d\mathbf{u}}{dt}(x,t) = \mathbf{F}(\mathbf{u}, x, t) \tag{1}$$

where $t > 0$, $x \in \mathbf{D}$ and $\mathbf{F}$ is a nonlinear function. This PDE can be rewritten in terms of conservative variables (instead of primitive variables). Discretizing domain $\mathbf{D}$ leads to a system of ordinary differential equations which can be solved using time integrator schemes such as Runge-Kutta scheme. Eventually the partial differential equation reduces to a system of nonlinear equations for each time step. Consider one instance of such a system

$$\mathbf{R}(\mathbf{w}) = 0, \tag{2}$$

where $\mathbf{w} \in \mathbb{R}^N$ and $\mathbf{R} : \mathbb{R}^N \to \mathbb{R}^N$ is the nonlinear residual function. Note that components of $\mathbf{w}$ consist of values of mass, moments in x, y, z directions and energy at grid points. To get a reasonable accuracy, the number of discretization points can get very large, i.e. N can be very large, and solving this problem can be very expensive. Therefore, usually $\mathbf{w}$ is approximated with a small number of basis functions: $\mathbf{w} \approx \tilde{\mathbf{w}} := \mathbf{w}_0 + \mathbf{V}\hat{\mathbf{w}}$, where $\mathbf{V} = [v_1 \; \dots \; v_p]$, $p << N$. Then solving equation (2) reduces to solving

$$\mathbf{R}(\mathbf{w}_0 + \mathbf{V}\hat{\mathbf{w}}) = 0.$$

This equation can be solved by minimizing the 2-norm of the function:

$$\min_{\hat{\mathbf{w}} \in \mathbb{R}^p} ||\mathbf{R}(\mathbf{w_0} + \mathbf{V}\hat{\mathbf{w}})||_2^2. \tag{3}$$

Conservation laws in CFD equations are important for getting accurate results. Therefore, we can consider a constrained minimization problem and require the solutions to satisfy the conservation laws on the full domain $\mathbf{D}$. The conservation of the mass of the fluid in the domain $\mathbf{D}$ is described by the following equation:

$$\frac{\partial}{\partial t}\int_{\mathbf{D}} \rho(x,t)dx + \int_{\mathbf{\Gamma}} \rho(x,\tau) \sum_{i=1}^{i=3} u_i(x,\tau) n_i(x,\tau) dx d\tau = \int_{\mathbf{D}} \mathbf{Q}_\rho(x,t)dx,$$

where $\mathbf{\Gamma}$ is the boundary of $\mathbf{D}$, $\mathbf{u} = [u_1, u_2, u_3]$ is the velocity of the fluid in 3D, ρ is the density function, $\mathbf{Q}_\rho$ is the force term and $\mathbf{n} = [n_1, n_2, n_3]$ is the normal component of the velocity. Further in the notes the Einstein notation will be used and the summation sign will be skipped:

$$\frac{\partial}{\partial t}\int_{\mathbf{D}} \rho(x,t)dx + \int_{\mathbf{\Gamma}} \rho(x,\tau) u_i(x,\tau) n_i(x,\tau) dx d\tau = \int_{\mathbf{D}} \mathbf{Q}_\rho(x,t)dx, \tag{4}$$

Similarly, the conservation of the moments and energy are described by the following equations:

$$\frac{\partial}{\partial t}\int_{\mathbf{D}} \rho(x,t) u_k(x,t)dx + \int_{\mathbf{\Gamma}} u_k(x,\tau)\rho(x,\tau) u_i(x,\tau) n_i(x,\tau) dx d\tau = \int_{\mathbf{D}} \mathbf{Q}_{\rho u}(x,t)dx, \quad k = 1,2,3 \tag{5}$$

$$\frac{\partial}{\partial t}\int_{\mathbf{D}} e(x,t)dx + \int_{\mathbf{\Gamma}} e(x,\tau)n_i(x,\tau)u_i(x,\tau)dxd\tau = \int_{\mathbf{D}} \mathbf{Q}_e(x,t)dx, \tag{6}$$

Thus, we can solve a constrained optimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^N} ||\mathbf{R}(\mathbf{w_0} + \mathbf{V}\hat{\mathbf{w}})||_2^2 \qquad \text{subject to conservation laws (4), (5), (6).}$$

Note that in 3D case we have only five constraints when conservation laws are required to hold on the full domain $\mathbf{D}$. However, taking any partition of the domain $\mathbf{D}$, we can require the conservation laws to hold on each subdomain. This will increase the number of constraints imposed to the problem. In other words, suppose $\mathbf{D} = \cup_{j=1}^{L}\mathbf{D}_j$, where $\mathrm{int}(\mathbf{D}_i)\cap\mathrm{int}(\mathbf{D}_j) = \emptyset,\ i \neq j$. Denote the boundary of each subset $\mathbf{D}_j$ by $\mathbf{\Gamma}_j$. Thus, overall we will have $n_c = 5L$ constraints:

$$\frac{\partial}{\partial t}\int_{\mathbf{D_j}} \rho(x,t)dx + \int_{\mathbf{\Gamma_j}} \rho(x,\tau)u_i(x,\tau)n_i(x,\tau)dxd\tau = \int_{\mathbf{D_j}} \mathbf{Q}_\rho(x,t)dx, \tag{7}$$

$$\frac{\partial}{\partial t}\int_{\mathbf{D_j}} \rho(x,t)u_k(x,t)dx + \int_{\mathbf{\Gamma_j}} u_k(x,\tau)\rho(x,\tau)u_i(x,\tau)n_i(x,\tau)dxd\tau = \int_{\mathbf{D_j}} \mathbf{Q}_{\rho u}(x,t)dx, \quad k = 1,2,3 \tag{8}$$

$$\frac{\partial}{\partial t}\int_{\mathbf{D_j}} e(x,t)dx + \int_{\mathbf{\Gamma_j}} e(x,\tau)n_i(x,\tau)u_i(x,\tau)dxd\tau = \int_{\mathbf{D_j}} \mathbf{Q}_e(x,t)dx, \tag{9}$$

Now the optimization problem has more constraints:

$$\min_{\mathbf{w}\in\mathbb{R}^N} ||\mathbf{R}(\mathbf{w_0} + \mathbf{V}\hat{\mathbf{w}})||_2^2 \ \text{subject to } n_c \text{ number of conservation laws (7), (8), (9).} \tag{10}$$

Note, there are three different possibilities for n_c:

1. $n_c < p$; underdetermined system of equations for constraints; need to use regularized optimization to solve for $\hat{\mathbf{w}}$.

2. $n_c = p$; determined system of equations for constraints; unique solution for $\hat{\mathbf{w}}$

3. $n_c > p$; overdetermined system of equations for constraints; use least squares.

Constrained optimization problems can be solved, for example, using $fmincon$ from MATLAB optimization toolbox. Another method to solve this problem is mentioned in the paper by R. Zimmerman and A. Vendl, Reduced Order Modeling of Steady State Flows Subject to Aerodynamic Constraints. They call this method as constrained least-squares(LSQ)-ROM.

Below is the illustration of this idea.

Consider quasi-1-D Euler equation,

$$\frac{\partial U}{\partial t} + \frac{1}{S}\frac{\partial FS}{\partial x} = Q,$$

where

$$U = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \quad F = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ (e+p)u \end{bmatrix}, \quad Q = \begin{bmatrix} 0 \\ \frac{p}{S}\frac{\partial A}{\partial x} \\ 0 \end{bmatrix},$$

$$p = (\gamma - 1)\rho\epsilon, \quad \epsilon = \frac{e}{p} - \frac{1}{2}u^2, \quad S = S(x).$$

S is a function of x and describes the cross sectional area of the channel. The following table describes the methods used for solving this problem.

ROM	**GNAT**
Galerkin	regular GNAT
Petrov-Galerkin	LSQ-ROM with real constraints
LSQ-ROM	LSQ-ROM with approximated constraints
$fmincon$	$fmincon$ with real constraints
	$fmincon$ with approximated equality and inequality constraints

The red cells are methods that at some point produce negative physical quantities such as density and pressure.

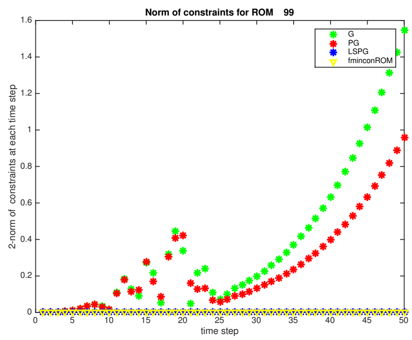
Conservation Laws on the Full Domain. This implies that in 1D case there are three constraints.

Surprisingly, the $fmincon$ and LSQ methods give almost the same answer for $\hat{w}$. In fact, for ROM when number of time steps is fifteen, $||\hat{w}_{fmincon} - \hat{w}_{LSQ}|| = O(10^{-8})$.

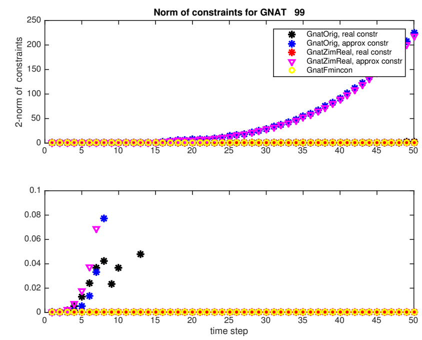
The comparison of methods are shown in the figures below. Here, $nstep = 50$, $nR = nJ = 25$ in GNAT algorithm. Three different cases for basis truncation are discussed in figures 1, 2 and 3: keeping 99%, 99.9% and 99.99% energy.

If we vary the number of basis vectors for truncation, then the norm of constraints for original and constrained GNAT can be found in figure (4).

Conservation Laws on Several Domains with Different Number of Basis Vectors. Results for the case with several domains is shown in figure 5 and figure 6. Probably we can change nR and nJ to get different results in future.
GNAT method with real constraints and Zimmerman method works when we have 25 time steps and the number of cells changes from 1 to 4 keeping 99.99% energy. When number of cells is 5, it produces negative physical quantities. If I increase number of basis vectors, then will work with 5 domain and will fail at 6th. My guess is that if I increase the number of basis vectors I can also increase the number of cells. The result for the norm of the constraint in this case is shown in figure 7



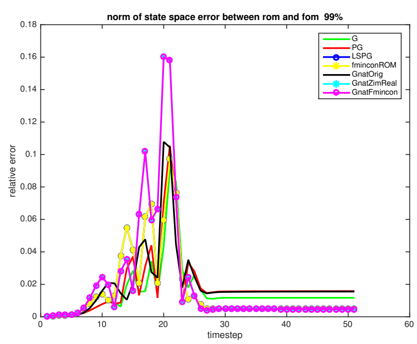
(a) norm of the constraint from ROM at each time step



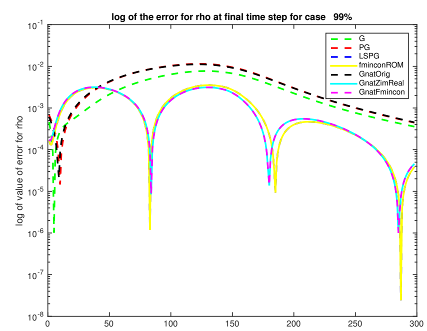
(b) norm of the constraint from Gnat at each time step, bottom figure is top figure zoomed

```
Frob norm of constraines for Galerkin      3.9008
Frob norm of constraines for PG     2.5206
Frob norm of constraines for LSPG     2.5701e-12
Frob norm of constraines for ROMfmincon     2.3943e-10
Frob norm of constraines for GNAT_original, real constr     5.213
Frob norm of constraines for GNAT_original, approx constr     552.7352
Frob norm of constraines for GNAT_Zim with realConstr, real constr     3.2425e-12
Frob norm of constraines for GNAT_Zim with realConstr, approx constr     536.3517
Frob norm of constraines for GNAT_fmincon real constrs     2.0425e-12
```

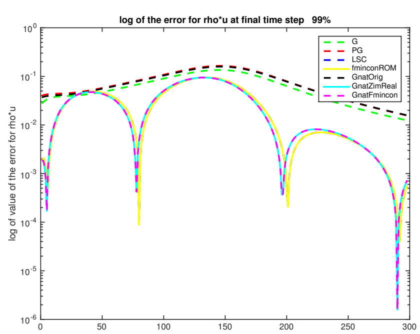
(c) Frob norm of the constraint for each method



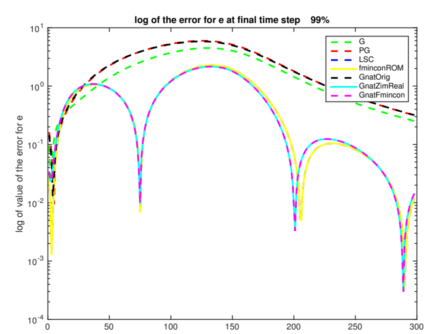
(d) see title



(e) log of the error for ρ

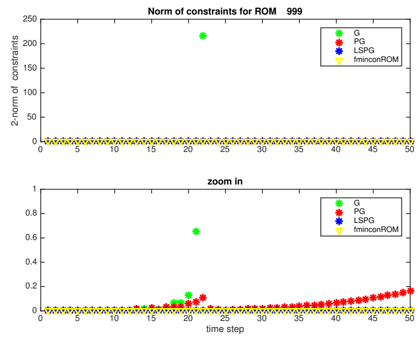


(f) log of the error for ρu

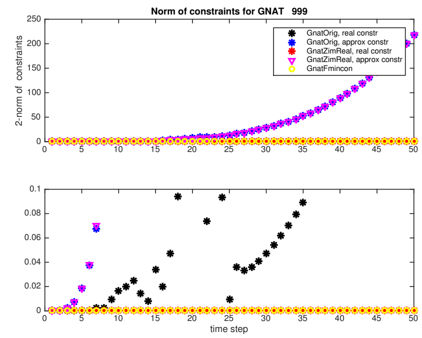


(g) log of the error for e

Figure 1: Time steps =50; 99% energy



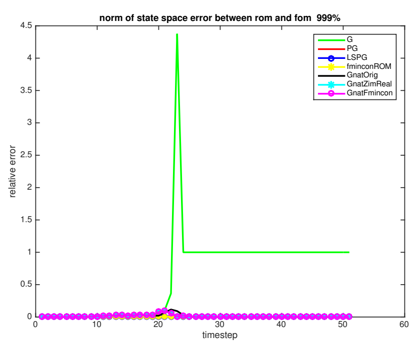
(a) norm of the constraint from ROM at each time step, bottom figure is the top figure zoomed



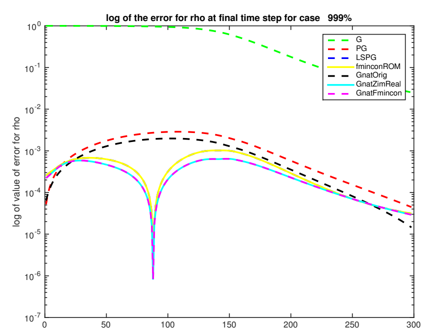
(b) norm of the constraint from Gnat at each time step, bottom figure is the top figure zoomed

```
Frob norm of constraines for Galerkin      216.1714
Frob norm of constraines for PG     0.43218
Frob norm of constraines for LSPG     3.2825e-13
Frob norm of constraines for ROMfmincon     2.0271e-13
Frob norm of constraines for GNAT_original, real constr     1.0774
Frob norm of constraines for GNAT_original, approx constr     537.5643
Frob norm of constraines for GNAT_Zim with realConstr, real constr     3.8208e-13
Frob norm of constraines for GNAT_Zim with realConstr, approx constr     537.7708
Frob norm of constraines for GNAT_fmincon real constrs     6.5103e-12
```

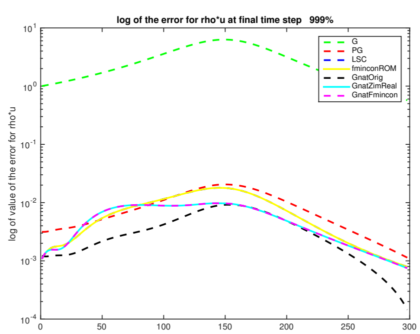
(c) Frob norm of the constraint for each method



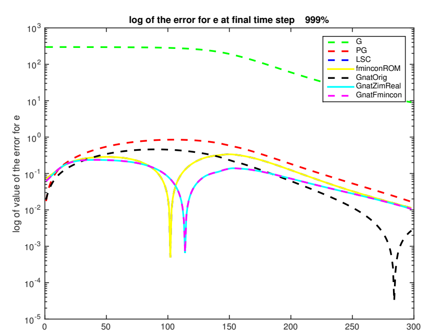
(d) see title



(e) log of the error for ρ

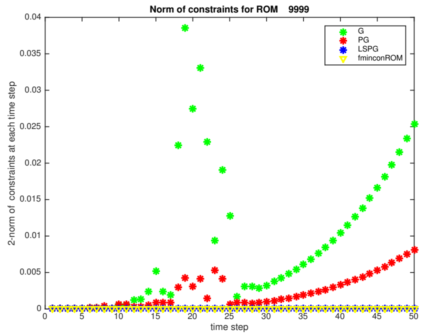


(f) log of the error for ρu

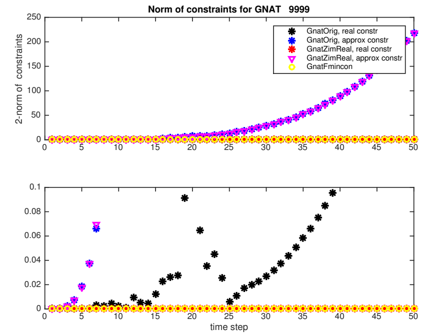


(g) log of the error for e

Figure 2: Time steps =50; 99.9% energy



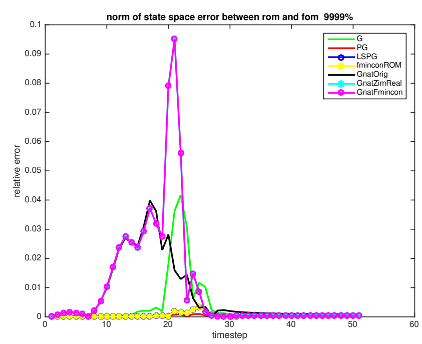
(a) norm of the constraint from ROM at each time step



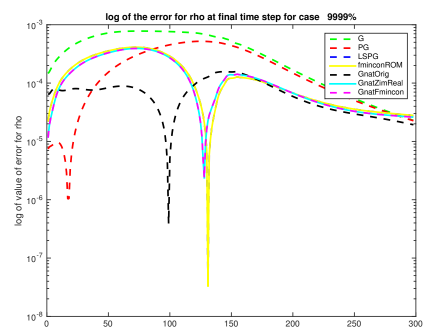
(b) norm of the constraint from Gnat at each time step, bottom figure is top figure zoomed

```
Frob norm of constraines for Galerkin      0.094453
Frob norm of constraines for PG     0.022388
Frob norm of constraines for LSPG     1.5465e-13
Frob norm of constraines for ROMfmincon     1.8926e-12
Frob norm of constraines for GNAT_original, real constr     0.69175
Frob norm of constraines for GNAT_original, approx constr     538.9416
Frob norm of constraines for GNAT_Zim with realConstr, real constr     4.0046e-13
Frob norm of constraines for GNAT_Zim with realConstr, approx constr     537.9605
Frob norm of constraines for GNAT_fmincon real constrs     4.5568e-13
```

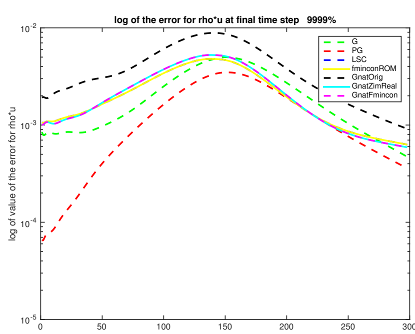
(c) Frob norm of the constraint for each method



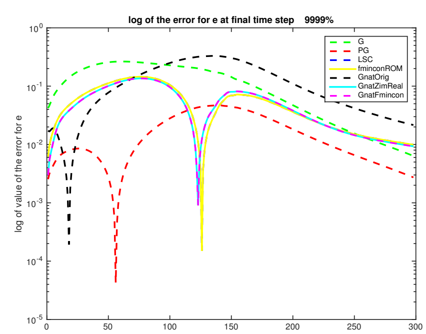
(d) see title



(e) log of the error for ρ

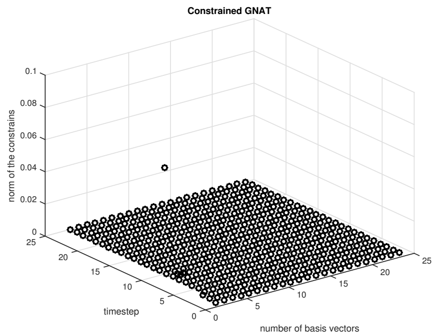


(f) log of the error for ρu

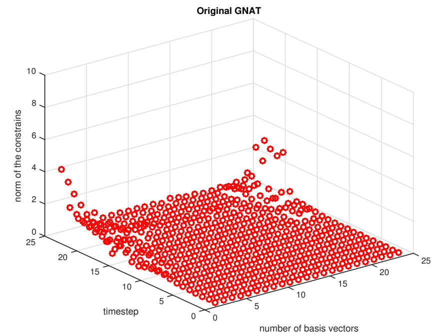


(g) log of the error for e

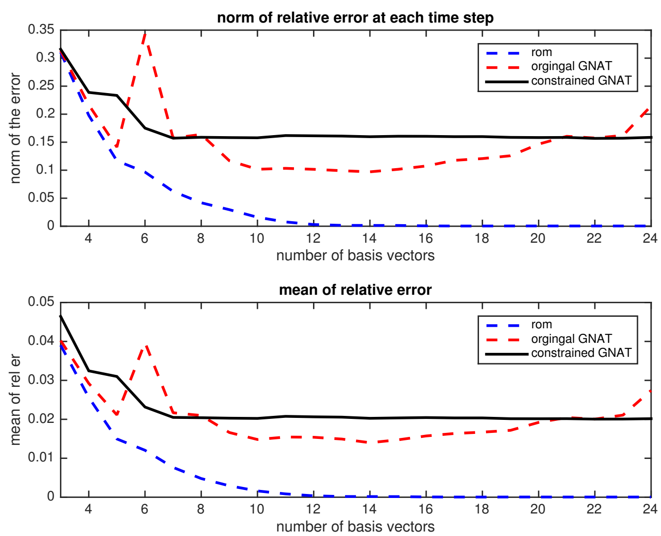
Figure 3: Time steps =50; 99.99% energy



(a) Norm of the constraints for constrained GNAT

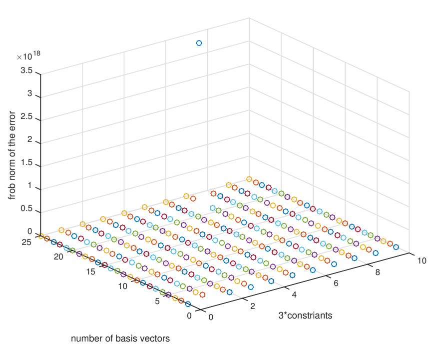


(b) Norm of the constraints for original GNAT

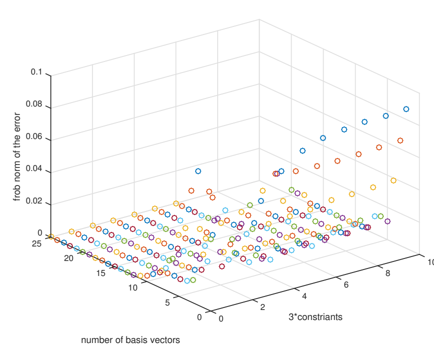


(c) Norm and the mean of relative error for different number of basis vectors

Figure 4: For 25 time steps and 3-24 basis vectors

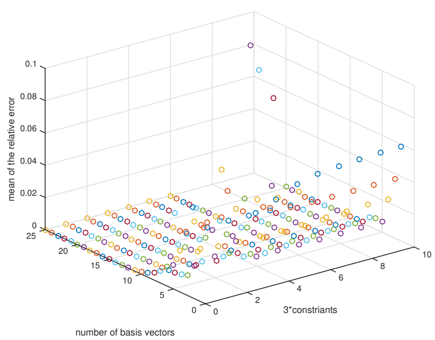


(a) Frobenius norm of the error divided by the frob norm of the fom soln

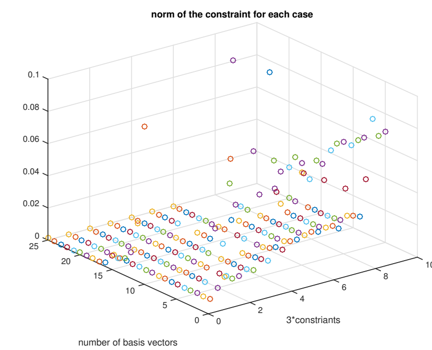


(b) Left plot zoomed

Figure 5: Time steps =25; realtive frobenius norm of the error fom-rom. 3*constr=0 corresponds to the original case with different basis numbers for truncation

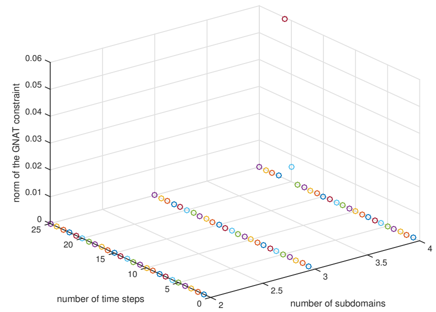


(a) mean of the relative error at each time step, zoomed in

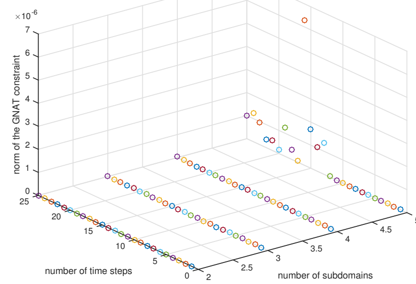


(b) norm of the constraints for each case, zoomed in

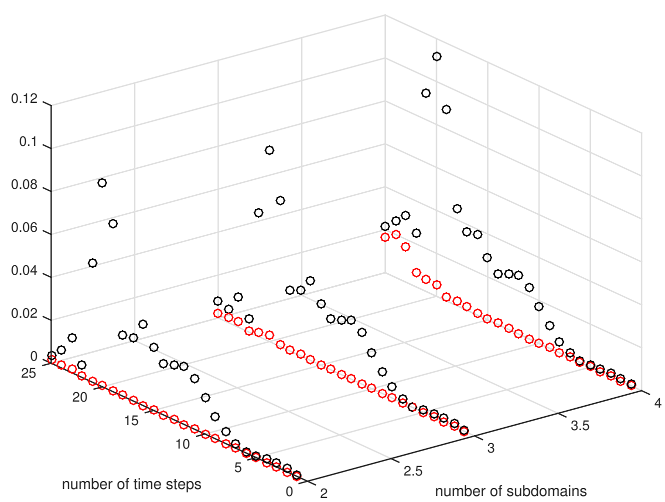
Figure 6: Time steps =25; different number of basis vectors and cells



(a) Norm of the constraint for GNAT, keep 99.99% energy(13 basis vectros)



(b) Norm of the constraint for GNAT, keep 20 basis vectors



(c) Relative error at each time step for different number of cells keeping 99.99% energy . Red is ROM, black is GNAT

Figure 7: timestep=25, different number of cells