Solutions of Polynomial Systems Derived from the Steady Cavity Flow Problem

Martin Mevissen\textsuperscript{1}  Kosuke Yokoyama\textsuperscript{2}  Nobuki Takayama\textsuperscript{2}

\textsuperscript{1}Tokyo Institute of Technology
\textsuperscript{2}Kobe University

ISSAC 2009, Seoul, Korea
The steady cavity flow problem

\[ 0 = \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} + \omega \quad \forall (x, y) \in [0, 1]^2, \]

\[ 0 = \frac{\partial \psi}{\partial x} \frac{\partial \omega}{\partial y} - \frac{\partial \psi}{\partial y} \frac{\partial \omega}{\partial x} + \frac{1}{R} \left( \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right) \quad \forall (x, y) \in [0, 1]^2. \]

Parameter \( R \) denotes the Reynolds number.

\[ v_1 = \frac{\partial \psi}{\partial y}, \quad v_2 = -\frac{\partial \psi}{\partial x} \]
The steady cavity flow problem

\[ 0 = \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} + \omega \quad \forall (x, y) \in [0, 1]^2, \]

\[ 0 = \frac{\partial \psi}{\partial x} \frac{\partial \omega}{\partial y} - \frac{\partial \psi}{\partial y} \frac{\partial \omega}{\partial x} + \frac{1}{R} \left( \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right) \quad \forall (x, y) \in [0, 1]^2. \]

\[ v_1 = \frac{\partial \psi}{\partial y}, \quad v_2 = -\frac{\partial \psi}{\partial x} \]

Parameter \( R \) denotes the Reynolds number.
The steady cavity flow problem

\[ 0 = \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} + \omega \quad \forall (x, y) \in [0, 1]^2, \]

\[ 0 = \frac{\partial \psi}{\partial x} \frac{\partial \omega}{\partial y} - \frac{\partial \psi}{\partial y} \frac{\partial \omega}{\partial x} + \frac{1}{R} \left( \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right) \quad \forall (x, y) \in [0, 1]^2. \]

Parameter \( R \) denotes the Reynolds number.

**Discretize** the steady cavity flow problem via a finite difference scheme!
Discretization yields

Discrete steady cavity flow problem DSCF(R,N)

\[-4\psi_{i,j} + \psi_{i+1,j} + \psi_{i-1,j} + \psi_{i,j+1} + \psi_{i,j-1} + h^2\omega_{i,j} = 0 \quad (i,j = 2, \ldots, N-1)\]
\[-4\omega_{i,j} + \omega_{i+1,j} + \omega_{i-1,j} + \omega_{i,j+1} + \omega_{i,j-1} + \frac{R}{4}(\psi_{i+1,j} - \psi_{i-1,j})(\omega_{i,j+1} - \omega_{i,j-1}) \quad = 0 \quad (i,j = 2, \ldots, N-1)\]
\[-\frac{R}{4}(\psi_{i,j+1} - \psi_{i,j-1})(\omega_{i+1,j} - \omega_{i-1,j}) = 0 \quad (i,j = 1, \ldots, N)\]
\[
\begin{align*}
\psi_{1,j} &= \psi_{N,j} = \psi_{i,1} = \psi_{i,N} \\
\omega_{1,j} &= -2\frac{\psi_{2,j}}{h^2}, \\
\omega_{N,j} &= -2\frac{\psi_{N-1,j}}{h^2}, \\
\omega_{i,1} &= -2\frac{\psi_{i,2}}{h^2}, \\
\omega_{i,N} &= -2\frac{\psi_{i,N-1+h}}{h^2}
\end{align*}
\]

Problem with two parameters $R$ and $N$ and dimension $n = 2(N - 2)^2$. 
Discretization yields

Discrete steady cavity flow problem DSCF(R,N)

\[\begin{align*}
-4\psi_{i,j} + \psi_{i+1,j} + \psi_{i-1,j} + \psi_{i,j+1} + \psi_{i,j-1} + h^2\omega_{i,j} &= 0 \quad (i, j = 2, \ldots, N-1) \\
-4\omega_{i,j} + \omega_{i+1,j} + \omega_{i-1,j} + \omega_{i,j+1} + \omega_{i,j-1} + \frac{R}{4}(\psi_{i+1,j} - \psi_{i-1,j})(\omega_{i,j+1} - \omega_{i,j-1}) + \\
&\quad + \frac{R}{4}(\psi_{i,j+1} - \psi_{i,j-1})(\omega_{i+1,j} - \omega_{i-1,j}) &= 0 \quad (i, j = 2, \ldots, N-1) \\
\psi_{1,j} = \psi_{N,j} = \psi_{i,1} = \psi_{i,N}
\end{align*}\]

\[\begin{align*}
\omega_{1,j} &= -2\frac{\psi_{2,j}}{h^2}, \\
\omega_{N,j} &= -2\frac{\psi_{N-1,j}}{h^2}, \\
\omega_{i,1} &= -2\frac{\psi_{i,2}}{h^2}, \\
\omega_{i,N} &= -2\frac{\psi_{i,N-1+h}}{h^2}
\end{align*}\]

Problem with two parameters \( R \) and \( N \) and dimension \( n = 2(N - 2)^2 \).
Discretization yields

Discrete steady cavity flow problem DSCF(R,N)

\[-4\psi_{i,j} + \psi_{i+1,j} + \psi_{i-1,j} + \psi_{i,j+1} + \psi_{i,j-1} + h^2 \omega_{i,j}\]

\[-4\omega_{i,j} + \omega_{i+1,j} + \omega_{i-1,j} + \omega_{i,j+1} + \omega_{i,j-1} + \frac{R}{4}(\psi_{i+1,j} - \psi_{i-1,j})(\omega_{i,j+1} - \omega_{i,j-1})\]

\[= 0 \quad (i, j = 2, \ldots, N-1)\]

\[-\frac{R}{4}(\psi_{i,j+1} - \psi_{i,j-1})(\omega_{i+1,j} - \omega_{i-1,j})\]

\[= 0 \quad (i, j = 1, \ldots, N)\]

\[\psi_{1,j} = \psi_{N,j} = \psi_{i,1} = \psi_{i,N}\]

\[\omega_{1,j} = -2\frac{\psi_{2,j}}{h^2}, \quad \omega_{N,j} = -2\frac{\psi_{N-1,j}}{h^2}\]

\[\omega_{i,1} = -2\frac{\psi_{i,2}}{h^2}, \quad \omega_{i,N} = -2\frac{\psi_{i,N-1+h}}{h^2}\]

Problem with two parameters \(R\) and \(N\) and dimension \(n = 2(N - 2)^2\).

Conjecture

DSCF\((R, N)\) has finitely many complex solutions for any \(R\) and \(N\).
Discretization yields

Discrete steady cavity flow problem DSCF(R,N)

\[-4\psi_{i,j} + \psi_{i+1,j} + \psi_{i-1,j} + \psi_{i,j+1} + \psi_{i,j-1} + h^2\omega_{i,j} = 0 \quad (i,j = 2, \ldots, N-1)\]

\[-4\omega_{i,j} + \omega_{i+1,j} + \omega_{i-1,j} + \omega_{i,j+1} + \omega_{i,j-1} + \frac{R}{4}(\psi_{i+1,j} - \psi_{i-1,j})(\omega_{i,j+1} - \omega_{i,j-1}) - \frac{R}{4}(\psi_{i,j+1} - \psi_{i,j-1})(\omega_{i+1,j} - \omega_{i-1,j}) = 0 \quad (i,j = 2, \ldots, N-1)\]

\[\psi_1,j = \psi_{N,j} = \psi_{i,1} = \psi_{i,N}\]

\[\omega_1,j = -2\frac{\psi_{2,j}}{h^2}, \quad \omega_{N,j} = -2\frac{\psi_{N-1,j}}{h^2}\]

\[\omega_{i,1} = -2\frac{\psi_{i,2}}{h^2}, \quad \omega_{i,N} = -2\frac{\psi_{i,N-1+h}}{h^2}\]

Problem with two parameters \(R\) and \(N\) and dimension \(n = 2(N - 2)^2\).

Conjecture

DSCF\((R, N)\) has finitely many complex solutions for any \(R\) and \(N\).
Discretization yields

Discrete steady cavity flow problem DSCF(R,N)

\[-4\psi_{i,j} + \psi_{i+1,j} + \psi_{i-1,j} + \psi_{i,j+1} + \psi_{i,j-1} + h^2\omega_{i,j} = 0 \quad (i, j = 2, \ldots, N-1)\]

\[-4\omega_{i,j} + \omega_{i+1,j} + \omega_{i-1,j} + \omega_{i,j+1} + \omega_{i,j-1} + \frac{R}{4}(\psi_{i+1,j} - \psi_{i-1,j})(\omega_{i,j+1} - \omega_{i,j-1}) + \frac{R}{4}(\psi_{i,j+1} - \psi_{i,j-1})(\omega_{i+1,j} - \omega_{i-1,j}) = 0 \quad (i, j = 2, \ldots, N-1)\]

\[\psi_{1,j} = \psi_{N,j} = \psi_{1,1} = \psi_{i,N}\]

\[\omega_{1,j} = -2\frac{\psi_{2,j}}{h^2}, \quad \omega_{N,j} = -2\frac{\psi_{N-1,j}}{h^2}\]

\[\omega_{i,1} = -2\frac{\psi_{i,2}}{h^2}, \quad \omega_{i,N} = -2\frac{\psi_{i,N-1} + h}{h^2}\]

Problem with two parameters $R$ and $N$ and dimension $n = 2(N - 2)^2$.

Conjecture

DSCF($R$, $N$) has finitely many complex solutions for any $R$ and $N$.

Aim: Enumerate all solutions of DSCF($R$, $N$) w.r.t. kinetic energy!
Outline

1. The discrete steady cavity flow problem

2. Sparse SDP relaxation method
   - Improving the accuracy of SDPR
   - Gröbner basis method and SDPR
   - Enumeration algorithm
   - Numerical results

3. Relations of Reynolds number $R$ and $CF(R, N)$
Idea of our approach:

Apply the **sparse semidefinite program relaxations (SDPR)** to a polynomial optimization problem (POP) derived from DSCF\((R, N)\).
Sparse SDP relaxations for POP

**Idea of our approach:**
Apply the **sparse semidefinite program relaxations (SDPR)** to a polynomial optimization problem (POP) derived from DSCF($R, N$).

**An SDP in standard form**

(P) \( \min \langle A_0, X \rangle \)

\[
\text{s.t.} \quad \langle A_i, X \rangle = b_i \quad (i = 1, \ldots, k) \\
X \succeq 0 \quad (X \text{ positive semidefinite})
\]

(D) \( \max \sum_{i=1}^{k} b_i y_i \)

\[
\text{s.t.} \quad A_0 - \sum_{i=1}^{k} A_i y_i \succeq 0
\]

\(A_0, \ldots, A_k, X \in S^n, b, y \in \mathbb{R}^k, \quad \langle G, H \rangle := \sum_{i,j} G_{i,j} H_{i,j} \).
Sparse SDP relaxations for POP

Idea of our approach:
Apply the \textit{sparse semidefinite program relaxations (SDPR)} to a polynomial optimization problem (POP) derived from DSCF($R$, $N$).

An SDP in standard form

\begin{align*}
(P) \quad \min & \quad \langle A_0, X \rangle \\
\text{s.t.} & \quad \langle A_i, X \rangle = b_i \quad (i = 1, \ldots, k) \\
& \quad X \succeq 0 \quad (X \text{ positive semidefinite})
\end{align*}

\begin{align*}
(D) \quad \max & \quad \sum_{i=1}^{k} b_i y_i \\
\text{s.t.} & \quad A_0 - \sum_{i=1}^{k} A_i y_i \succeq 0
\end{align*}

$A_0, \ldots, A_k, X \in \mathbb{S}^n$, $b, y \in \mathbb{R}^k$, $\langle G, H \rangle := \sum_{i,j} G_{i,j} H_{i,j}$.

- If there exists \textit{interior feasible} $(\bar{X}, \bar{y})$, $\min(P) = \max(D)$ holds.
Sparse SDP relaxations for POP

Idea of our approach:

Apply the **sparse semidefinite program relaxations (SDPR)** to a polynomial optimization problem (POP) derived from DSCF($R, N$).

An SDP in standard form

(P) \[
\begin{align*}
\min & \quad \langle A_0, X \rangle \\
s.t. & \quad \langle A_i, X \rangle = b_i \quad (i = 1, \ldots, k) \\
& \quad X \succeq 0 \\
\end{align*}
\]

(D) \[
\begin{align*}
\max & \quad \sum_{i=1}^{k} b_i y_i \\
s.t. & \quad A_0 - \sum_{i=1}^{k} A_i y_i \succeq 0 \\
\end{align*}
\]

If there exists **interior feasible** ($\bar{X}, \bar{y}$), $\min(P) = \max(D)$ holds. 

(SDP) can be solved in poly time [Khachian 1979], efficiently by primal-dual interior point method [Nesterov, Nemirovski 1994].
Sparse SDP relaxations for POP

\begin{align*}
\text{min} & \quad F(x) \\
\text{s.t.} & \quad g_j(x) \geq 0 \quad \forall \ j \in \{1, \ldots, k\}, \\
& \quad h_i(x) = 0 \quad \forall \ i \in \{1, \ldots, l\}.
\end{align*}
Sparse POP can be approximated by a **hierarchy of sparse SDP relaxations** $\text{SDPR}(w)$ [Lasserre, Waki, Kim, Kojima].

\[
\begin{align*}
\min \quad F(x) \\
\text{s.t.} \quad g_j(x) &\geq 0 \quad \forall \ j \in \{1, \ldots, k\}, \\
\quad h_i(x) &= 0 \quad \forall \ i \in \{1, \ldots, l\}.
\end{align*}
\]

- \[\min(\text{SDPR}(w)) \rightarrow \min(\text{POP}), \text{ for } w \rightarrow \infty,\]
- holds if compactness conditions for $\text{feas}(\text{POP})$ are satisfied.
Sparse POP can be approximated by a hierarchy of sparse SDP relaxations $\text{SDPR}(w)$ [Lasserre, Waki, Kim, Kojima].

\[
\min(\text{SDPR}(w)) \rightarrow \min(\text{POP}), \quad \text{for } w \rightarrow \infty,
\]
holds if compactness conditions for $\text{feas}(\text{POP})$ are satisfied.

POP derived from PDE satisfy structured sparsity patterns [Mevissen, Kojima, Nie, Takayama].
Sparse POP can be approximated by a hierarchy of sparse SDP relaxations SDPR(\(w\)) [Lasserre, Waki, Kim, Kojima].

\[
\begin{align*}
\min & \quad F(x) \\
\text{s.t.} & \quad g_j(x) \geq 0 \quad \forall \, j \in \{1, \ldots, k\}, \quad h_i(x) = 0 \quad \forall \, i \in \{1, \ldots, l\}.
\end{align*}
\]

\[
\min(\text{SDPR}(w)) \rightarrow \min(\text{POP}), \text{ for } w \rightarrow \infty,
\]

holds if compactness conditions for feas(POP) are satisfied.

POP derived from PDE satisfy structured sparsity patterns [Mevissen, Kojima, Nie, Takayama].

Take DSCF(R,N) as constraints and choose an objective to obtain a POP!
An objective is needed to derive a POP:

Kinetic energy of the flow: 
\[ \int \int_{[0,1]^2} \left( \left( \frac{\partial \psi}{\partial y} \right)^2 + \left( \frac{\partial \psi}{\partial x} \right)^2 \right) \, dx \, dy \]

Discretize kinetic energy:

\[ F(\psi, \omega) = \frac{1}{4} \sum_{2 \leq i,j \leq N-1} \left( \psi_{i+1,j} - \psi_{i-1,j} \right)^2 + \left( \psi_{i,j+1} - \psi_{i,j-1} \right)^2 \]
An objective is needed to derive a POP:

Kinetic energy of the flow: \[ \int \int_{[0,1]^2} \left( \left( \frac{\partial \psi}{\partial y} \right)^2 + \left( \frac{\partial \psi}{\partial x} \right)^2 \right) \, dx \, dy \]

Discretize kinetic energy:

\[ F(\psi, \omega) = \frac{1}{4} \sum_{2 \leq i, j \leq N-1} (\psi_{i+1,j} - \psi_{i-1,j})^2 + (\psi_{i,j+1} - \psi_{i,j-1})^2 \]

The cavity flow optimization problem \( \text{CF}(R, N) \)

\[
\begin{align*}
\text{min} & \quad F(\psi, \omega) \\
\text{s.t.} & \quad \text{DSCF}(R, N)
\end{align*}
\]
An objective is needed to derive a POP:

Kinetic energy of the flow:
\[
\int \int_{[0,1]^2} \left( \left( \frac{\partial \psi}{\partial y} \right)^2 + \left( \frac{\partial \psi}{\partial x} \right)^2 \right) \, dx \, dy
\]

Discretize kinetic energy:
\[
F(\psi, \omega) = \frac{1}{4} \sum_{2 \leq i, j \leq N-1} \left( \psi_{i+1,j} - \psi_{i-1,j} \right)^2 + \left( \psi_{i,j+1} - \psi_{i,j-1} \right)^2
\]

The cavity flow optimization problem \( CF(R, N) \)

\[
\begin{align*}
\text{min} & \quad F(\psi, \omega) \\
\text{s.t.} & \quad DSCF(R, N)
\end{align*}
\]

Proposition
An objective is needed to derive a POP:

Kinetic energy of the flow:

\[ \int \int_{[0,1]^2} \left( \left( \frac{\partial \psi}{\partial y} \right)^2 + \left( \frac{\partial \psi}{\partial x} \right)^2 \right) \, dx \, dy \]

Discretize kinetic energy:

\[ F(\psi, \omega) = \frac{1}{4} \sum_{2 \leq i, j \leq N-1} \left( \psi_{i+1,j} - \psi_{i-1,j} \right)^2 + \left( \psi_{i,j+1} - \psi_{i,j-1} \right)^2 \]

The cavity flow optimization problem CF(R, N)

\[ \text{min} \quad F(\psi, \omega) \]
\[ \text{s.t.} \quad \text{DSCF}(R, N) \]

Proposition

\( a) \quad \text{CF}(0, N) \text{ is a convex quadratic for any } N. \)
An objective is needed to derive a POP:

Kinetic energy of the flow:
\[ \int \int_{[0,1]^2} \left( (\frac{\partial \psi}{\partial y})^2 + (\frac{\partial \psi}{\partial x})^2 \right) \, dx \, dy \]

Discretize kinetic energy:
\[ F(\psi, \omega) = \frac{1}{4} \sum_{2 \leq i,j \leq N-1} (\psi_{i+1,j} - \psi_{i-1,j})^2 + (\psi_{i,j+1} - \psi_{i,j-1})^2 \]

The cavity flow optimization problem \( \text{CF}(R, N) \)

\[ \min \quad F(\psi, \omega) \]
\[ \text{s.t.} \quad \text{DSCF}(R, N) \]

Proposition
a) \( \text{CF}(0, N) \) is a **convex quadratic** for any \( N \).
b) \( \text{CF}(R, N) \) is **non-convex quadratic** for any \( N \), if \( R \neq 0 \).
It can be shown that CF($R$, $N$) satisfies a some structured sparsity pattern.

Figure: Chordal correlative sparsity pattern matrix for CF($R$, 20)
It can be shown that CF($R, N$) satisfies a some **structured sparsity** pattern.

**Figure:** Chordal correlative sparsity pattern matrix for CF($R, 20$)

Therefore, the **sparse SDP relaxations** [Waki et al.] are **efficient** to solve CF($R, N$)!
It can be shown that $\text{CF}(R, N)$ satisfies a some structured sparsity pattern.

Therefore, the sparse SDP relaxations [Waki et al.] are efficient to solve $\text{CF}(R, N)$!

We will apply $\text{SDPR}(w)$ with order $w \in \{1, 2\}$. 

**Figure**: Chordal correlative sparsity pattern matrix for $\text{CF}(R, 20)$
Outline

1. The discrete steady cavity flow problem

2. Sparse SDP relaxation method
   - Improving the accuracy of SDPR
   - Gröbner basis method and SDPR
   - Enumeration algorithm
   - Numerical results

3. Relations of Reynolds number $R$ and $\text{CF}(R, N)$
How to improve the accuracy of SDPR

Problem

SDPR(1) and SDPR(2) may not yield accurate approximation to the global optimizers of CF($R, N$).
How to improve the accuracy of SDPR

**Problem**

SDPR(1) and SDPR(2) may not yield accurate approximation to the global optimizers of CF($R$, $N$).

**Technique 1**

Impose **tight lower and upper bounds** for all $\psi_i$ and $\omega_i$,

\[
\text{lbd}_i^\psi \leq \psi_i \leq \text{ubd}_i^\psi \quad \text{and} \quad \text{lbd}_i^\omega \leq \omega_i \leq \text{ubd}_i^\omega \quad \forall \ 1 \leq i \leq N^2.
\]
How to improve the accuracy of SDPR

Problem
SDPR(1) and SDPR(2) may not yield accurate approximation to the global optimizers of CF(R, N).

Technique 1
Impose tight lower and upper bounds for all $\psi_i$ and $\omega_i$,

$$\text{lbd}_i^\psi \leq \psi_i \leq \text{ubd}_i^\psi \quad \text{and} \quad \text{lbd}_i^\omega \leq \omega_i \leq \text{ubd}_i^\omega \quad \forall \; 1 \leq i \leq N^2.$$ 

Technique 2
Apply locally convergent optimization techniques like Newton’s method for nonlinear systems or sequential quadratic programming (SQP) starting from the SDPR(w) solution.
Combining these techniques yields

**The SDPR method**
Combining these techniques yields

The SDPR method

1. Choose the two parameters $R$ and $N$. 
Combining these techniques yields

The SDPR method

1. Choose the two parameters $R$ and $N$.
2. Apply SDPR($w$) to CF($R, N$) and obtain solution $\tilde{u} := (\tilde{\psi}, \tilde{\omega})$. 

Mevissen (Tokyo Tech) Solutions for Polynomial Systems ISSAC 2009 11 / 26
Combining these techniques yields

**The SDPR method**

1. Choose the two parameters $R$ and $N$.
2. Apply SDPR($w$) to CF($R, N$) and obtain solution $\tilde{u} := (\tilde{\psi}, \tilde{\omega})$.
3. Apply sequential quadratic programming (SQP) to CF($R, N$) or Newton’s method to DSCF($R, N$), each of them starting from $\tilde{u}$, and obtain $u := (\psi, \omega)$. 

Mevissen (Tokyo Tech) Solutions for Polynomial Systems ISSAC 2009 11 / 26
Outline

1. The discrete steady cavity flow problem

2. Sparse SDP relaxation method
   - Improving the accuracy of SDPR
   - Gröbner basis method and SDPR
   - Enumeration algorithm
   - Numerical results

3. Relations of Reynolds number $R$ and $CF(R, N)$
Use Gröbner basis method to tune SDPR method and validate its numerical results!
Use **Gröbner basis method** to tune SDPR method and validate its numerical results!

Apply **rational univariate representation (RUR)** to DSCF($R, 5$):

Find all complex solutions of a system with 18 variables, 9 appear as linear and 9 appear as quadratic variables. Then, all real solutions can be enumerated w.r.t. their discretized kinetic energy.
Use **Gröbner basis method** to tune SDPR method and validate its numerical results!

Apply **rational univariate representation (RUR)** to DSCF($R, 5$):

Find all complex solutions of a system with 18 variables, 9 appear as linear and 9 appear as quadratic variables. Then, all real solutions can be enumerated w.r.t. their discretized kinetic energy.

**Claim:**

Results obtained by RUR for $N = 5$ can be used for tuning SDPR method for $N > 5$, too!
1. The discrete steady cavity flow problem

2. Sparse SDP relaxation method
   - Improving the accuracy of SDPR
   - Gröbner basis method and SDPR
   - Enumeration algorithm
   - Numerical results

3. Relations of Reynolds number $R$ and $CF(R, N)$
Algorithm to enumerate all solutions of DSCF\((R, N)\)

Iterate the 5 steps to approximate the \(k\) smallest energy solutions:
Algorithm to enumerate all solutions of DSCF($R, N$)

Iterate the 5 steps to approximate the $k$ smallest energy solutions:

1. Given $u^{(k-1)}$, the approximation to the $(k - 1)$th smallest energy solution obtained by solving SDPR$^{k-1}(w)$. 

Mevissen (Tokyo Tech) Solutions for Polynomial Systems ISSAC 2009 15 / 26
Algorithm to enumerate all solutions of DSCF($R, N$)

Iterate the 5 steps to approximate the $k$ smallest energy solutions:

1. Given $u^{(k-1)}$, the approximation to the $(k-1)$th smallest energy solution obtained by solving SDPR$_{k-1}(w)$.

2. Choose $\epsilon_1^k, \epsilon_2^k > 0$ and integers $b_1^k, b_2^k \in \{1, \ldots, (N-2)^2\}$. 

Algorithm to enumerate all solutions of DSCF($R, N$)

Iterate the 5 steps to approximate the $k$ smallest energy solutions:

1. Given $u^{(k-1)}$, the approximation to the $(k - 1)$th smallest energy solution obtained by solving SDPR$^{k-1}(w)$.
2. Choose $\epsilon_k^1, \epsilon_k^2 > 0$ and integers $b_k^1, b_k^2 \in \{1, \ldots, (N - 2)^2\}$.
3. Add the following quadratic constraints to SDPR$^{k-1}(w)$ and denote the resulting (tighter) SDP relaxation as SDPR$^k(w)$.

\[
(u_j - u_j^{(k-1)})^2 \geq \epsilon_k^1 \quad \forall 1 \leq j \leq b_k^1, \\
(u_j + (N-2)^2 - u_j^{(k-1)})^2 \geq \epsilon_k^2 \quad \forall 1 \leq j \leq b_k^2.
\]
Algorithm to enumerate all solutions of DSCF($R,N$)

Iterate the 5 steps to approximate the $k$ smallest energy solutions:

1. Given $u^{(k-1)}$, the approximation to the $(k-1)$th smallest energy solution obtained by solving SDPR$^{k-1}(w)$.

2. Choose $\epsilon_1^k, \epsilon_2^k > 0$ and integers $b_1^k, b_2^k \in \{1, \ldots, (N-2)^2\}$.

3. Add the following quadratic constraints to SDPR$^{k-1}(w)$ and denote the resulting (tighter) SDP relaxation as SDPR$^k(w)$.

\begin{align}
(u_j - u_j^{(k-1)})^2 &\geq \epsilon_1^k & \forall 1 \leq j \leq b_1^k, \\
(u_{j+(N-2)^2} - u_{j+(N-2)^2}^{(k-1)})^2 &\geq \epsilon_2^k & \forall 1 \leq j \leq b_2^k.
\end{align}

4. Solve SDPR$^k(w)$ and obtain a first approximation $\tilde{u}^{(k)}$. 

Mevissen (Tokyo Tech) Solutions for Polynomial Systems ISSAC 2009 15 / 26
Algorithm to enumerate all solutions of DSCF\((R, N)\)

Iterate the 5 steps to approximate the \(k\) smallest energy solutions:

1. Given \(u^{(k-1)}\), the approximation to the \((k-1)\)th smallest energy solution obtained by solving SDPR\(^{(k-1)}(w)\).

2. Choose \(\epsilon_1^k, \epsilon_2^k > 0\) and integers \(b_1^k, b_2^k \in \{1, \ldots, (N-2)^2\}\).

3. Add the following quadratic constraints to SDPR\(^{(k-1)}(w)\) and denote the resulting (tighter) SDP relaxation as SDPR\(^k(w)\).

\[
\begin{align*}
(u_j - u_j^{(k-1)})^2 &\geq \epsilon_1^k \quad \forall 1 \leq j \leq b_1^k, \\
(u_j + (N-2)^2 - u_j^{(k-1)})^2 &\geq \epsilon_2^k \quad \forall 1 \leq j \leq b_2^k.
\end{align*}
\]

4. Solve SDPR\(^k(w)\) and obtain a first approximation \(\tilde{u}^{(k)}\).

5. Apply Newton’s method or SQP with \(\tilde{u}^{(k)}\) as starting point. Obtain \(u^{(k)}\) as an approximation to \(u^{(k)*}\).
Algorithm to enumerate all solutions of DSCF\((R, N)\)

Iterate the 5 steps to approximate the \(k\) smallest energy solutions:

1. **Given** \(u^{(k-1)}\), the approximation to the \((k - 1)\)th smallest energy solution obtained by solving SDPR\(^{k-1}\)(\(w\)).
2. Choose \(\epsilon^k_1, \epsilon^k_2 > 0\) and integers \(b^k_1, b^k_2 \in \{1, \ldots, (N - 2)^2\}\).
3. Add the following quadratic constraints to SDPR\(^{k-1}\)(\(w\)) and denote the resulting (tighter) SDP relaxation as SDPR\(^k\)(\(w\)).

\[
\begin{align*}
(u_j - u_j^{(k-1)})^2 &\geq \epsilon^k_1 & \forall 1 \leq j \leq b^k_1, \\
(u_j + (N-2)^2 - u_j^{(k-1)})^2 &\geq \epsilon^k_2 & \forall 1 \leq j \leq b^k_2. 
\end{align*}
\]

4. Solve SDPR\(^k\)(\(w\)) and obtain a first approximation \(\tilde{u}^{(k)}\).
5. Apply Newton’s method or SQP with \(\tilde{u}^{(k)}\) as starting point. Obtain \(u^{(k)}\) as an approximation to \(u^{(k)\ast}\).
Algorithm to enumerate all solutions of DSCF\((R, N)\)

Iterate the 5 steps to approximate the \(k\) smallest energy solutions:

1. Given \(u^{(k-1)}\), the approximation to the \((k - 1)\)th smallest energy solution obtained by solving SDPR\(^{k-1}(w)\).

2. Choose \(\epsilon_1^k, \epsilon_2^k > 0\) and integers \(b_1^k, b_2^k \in \{1, \ldots, (N - 2)^2\}\).

3. Add the following quadratic constraints to SDPR\(^{k-1}(w)\) and denote the resulting (tighter) SDP relaxation as SDPR\(^k(w)\).

\[
\begin{align*}
(u_j - u_j^{(k-1)})^2 &\geq \epsilon_1^k & \forall 1 \leq j \leq b_1^k, \\
(u_j + (N-2)^2 - u_j^{(k-1)})^2 &\geq \epsilon_2^k & \forall 1 \leq j \leq b_2^k.
\end{align*}
\tag{1}
\]

4. Solve SDPR\(^k(w)\) and obtain a first approximation \(\tilde{u}^{(k)}\).

5. Apply Newton’s method or SQP with \(\tilde{u}^{(k)}\) as starting point. Obtain \(u^{(k)}\) as an approximation to \(u^{(k)*}\).

Constraints (1) have shown to be superior to alternatives.
Proposition 2

Let $R$ and $N$ be fixed, $(u^{(1)}, \ldots, u^{(k-1)})$ be the output of the first $(k - 1)$ iterations. If this output is a sufficiently close approximation of $(u^{(1)}\star, \ldots, u^{(k-1)}\star)$, and if $\text{DSCF}(R, N)$ is finite and distinct in terms of $F$, i.e. $F(u^{(1)}\star) < F(u^{(2)}\star) < \ldots$, then there exist $b \in \{1, \ldots, n\}$ and $\epsilon \in \mathbb{R}^b$ such that $u^{(k)}$ from the $k$th iteration satisfies

$$u^{(k)}(w) \rightarrow u^{(k)}\star \text{ when } w \rightarrow \infty.$$
Convergence result for the enumeration algorithm

Proposition 2

Let $R$ and $N$ be fixed, $(u^{(1)}, \ldots, u^{(k-1)})$ be the output of the first $(k - 1)$ iterations. If this output is a sufficiently close approximation of $(u^{(1)*}, \ldots, u^{(k-1)*})$, and if $\text{DSCF}(R, N)$ is finite and distinct in terms of $F$, i.e. $F(u^{(1)*}) < F(u^{(2)*}) < \ldots$, then there exist $b \in \{1, \ldots, n\}$ and $\epsilon \in \mathbb{R}^b$ such that $u^{(k)}$ from the $k$th iteration satisfies

$$u^{(k)}(w) \to u^{(k)*} \quad \text{when } w \to \infty.$$

Proof: Apply of Lasserre’s theorem to sequence of POPs generated by the algorithm.
Proposition 2

Let $R$ and $N$ be fixed, $(u^{(1)}, \ldots, u^{(k-1)})$ be the output of the first $(k - 1)$ iterations. If this output is a sufficiently close approximation of $(u^{(1)}\ast, \ldots, u^{(k-1)}\ast)$, and if $DSCF(R, N)$ is finite and distinct in terms of $F$, i.e. $F(u^{(1)}\ast) < F(u^{(2)}\ast) < \ldots$, then there exist $b \in \{1, \ldots, n\}$ and $\epsilon \in \mathbb{R}^b$ such that $u^{(k)}$ from the $k$th iteration satisfies

$$u^{(k)}(w) \rightarrow u^{(k)}\ast \quad \text{when } w \rightarrow \infty.$$ 

Proof: Apply of Lasserre’s theorem to sequence of POPs generated by the algorithm.

But:

In numerical experiments we are restricted to $w \in \{1, 2\}$. 
Proposition 2

Let $R$ and $N$ be fixed, $(u^{(1)}, \ldots, u^{(k-1)})$ be the output of the first $(k - 1)$ iterations. If this output is a sufficiently close approximation of $(u^{(1)*}, \ldots, u^{(k-1)*})$, and if $\text{DSCF}(R, N)$ is finite and distinct in terms of $F$, i.e. $F(u^{(1)*}) < F(u^{(2)*}) < \ldots$, then there exist $b \in \{1, \ldots, n\}$ and $\epsilon \in \mathbb{R}^b$ such that $u^{(k)}$ from the $k$th iteration satisfies

$$u^{(k)}(w) \to u^{(k)*} \quad \text{when} \quad w \to \infty.$$ 

Proof: Apply of Lasserre’s theorem to sequence of POPs generated by the algorithm.

But:

In numerical experiments we are restricted to $w \in \{1, 2\}$. 
Convergence result for the enumeration algorithm

Proposition 2

Let $R$ and $N$ be fixed, $(u^{(1)}, \ldots, u^{(k-1)})$ be the output of the first $(k - 1)$ iterations. If this output is a sufficiently close approximation of $(u^{(1)*}, \ldots, u^{(k-1)*})$, and if $\text{DSCF}(R, N)$ is finite and distinct in terms of $F$, i.e. $F(u^{(1)*}) < F(u^{(2)*}) < \ldots$, then there exist $b \in \{1, \ldots, n\}$ and $\epsilon \in \mathbb{R}^b$ such that $u^{(k)}$ from the $k$th iteration satisfies

$$u^{(k)}(w) \rightarrow u^{(k)*} \quad \text{when} \quad w \rightarrow \infty.$$ 

Proof: Apply of Lasserre’s theorem to sequence of POPs generated by the algorithm.

But:

In numerical experiments we are restricted to $w \in \{1, 2\}$.

Use results from RUR for $N = 5$ to tune $\epsilon$ and $b$ for $N \geq 5$. 

Mevissen (Tokyo Tech) Solutions for Polynomial Systems ISSAC 2009 16 / 26
Outline

1. The discrete steady cavity flow problem

2. Sparse SDP relaxation method
   - Improving the accuracy of SDPR
   - Gröbner basis method and SDPR
   - Enumeration algorithm
   - Numerical results

3. Relations of Reynolds number $R$ and $CF(R, N)$
Can verify by Gröbner basis method [RUR]:
\(u^{(0)}, u^{(1)}\) and \(u^{(2)}\) are indeed the 3 smallest energy solutions!
<table>
<thead>
<tr>
<th>$k$</th>
<th>$w$</th>
<th>$\epsilon^k_1$</th>
<th>$b^k_1$</th>
<th>$b^k_2$</th>
<th>$t_C$</th>
<th>$\epsilon_{sc}$</th>
<th>$F(u^{(k)})$</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>3e-7</td>
<td>3.4e-4</td>
<td>$u^{(0)}$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1e-3</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td>5e-4</td>
<td>3.4e-4</td>
<td>$u^{(0)}$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1e-6</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>6e-6</td>
<td>3.4e-4</td>
<td>$u^{(0)}$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1e-5</td>
<td>5</td>
<td>0</td>
<td>9</td>
<td>5e-6</td>
<td>5.9e-4</td>
<td>$u^{(1)}$</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1e-5</td>
<td>5</td>
<td>0</td>
<td>14</td>
<td>5e-6</td>
<td>5.2e-3</td>
<td>$u^{(2)}$</td>
</tr>
</tbody>
</table>

$CF(20000,7)$

$u^{(0)}$

$u^{(1)}$

$u^{(2)}$
Right choice for $\epsilon$ and $b$ crucial!

<table>
<thead>
<tr>
<th>$k$</th>
<th>$w$</th>
<th>$\epsilon_1^k$</th>
<th>$b_1^k$</th>
<th>$b_2^k$</th>
<th>$t_C$</th>
<th>$\epsilon_{sc}$</th>
<th>$F(u^{(k)})$</th>
<th>solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>3e-7</td>
<td>3.4e-4</td>
<td>$u^{(0)}$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1e-3</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td>5e-4</td>
<td>3.4e-4</td>
<td>$u^{(0)}$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1e-6</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>6e-6</td>
<td>3.4e-4</td>
<td>$u^{(0)}$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1e-5</td>
<td>5</td>
<td>0</td>
<td>9</td>
<td>5e-6</td>
<td>5.9e-4</td>
<td>$u^{(1)}$</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1e-5</td>
<td>5</td>
<td>0</td>
<td>14</td>
<td>5e-6</td>
<td>5.2e-3</td>
<td>$u^{(2)}$</td>
</tr>
<tr>
<td>$k$</td>
<td>$w$</td>
<td>$\epsilon_1^k$</td>
<td>$b_1^k$</td>
<td>$b_2^k$</td>
<td>$t_C$</td>
<td>$\epsilon_{sc}$</td>
<td>$F(u^{(k)})$</td>
<td>solution</td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
<td>----------------</td>
<td>--------</td>
<td>--------</td>
<td>------</td>
<td>--------------</td>
<td>--------------</td>
<td>----------</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>2e-7</td>
<td>3.4e-4</td>
<td>$u^{(0)}(1)$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>5e-6</td>
<td>5</td>
<td>0</td>
<td>7</td>
<td>6e-9</td>
<td>7.3e-4</td>
<td>$u^{(1)}(1)$</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>5e-6</td>
<td>5</td>
<td>0</td>
<td>11</td>
<td>3e-6</td>
<td>5.9e-4</td>
<td>$u^{(2)}(1)$</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>8e-6</td>
<td>5</td>
<td>0</td>
<td>16</td>
<td>5e-6</td>
<td>2.3e-4</td>
<td>$u^{(3)}(1)$</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>5872</td>
<td>8e-10</td>
<td>2.6e-4</td>
<td>$u^{(0)}(2)$</td>
</tr>
</tbody>
</table>

Enumeration algorithm more efficient in approximating global minimizer than classical SDP relaxation!
1. The discrete steady cavity flow problem

2. Sparse SDP relaxation method
   - Improving the accuracy of SDPR
   - Gröbner basis method and SDPR
   - Enumeration algorithm
   - Numerical results

3. Relations of Reynolds number $R$ and $\text{CF}(R, N)$
Observe: DSCF($R, N$) is more difficult to solve for larger $R$. 
Observe: DSCF($R, N$) is more difficult to solve for larger $R$.

Question: How does $u^*$ behave for increasing $R$?
Observe: DSCF(R, N) is more difficult to solve for larger R.

Question: How does $u^*$ behave for increasing $R$?
Observe: DSCF($R, N$) is more difficult to solve for larger $R$.

Question: How does $u^*$ behave for increasing $R$?

Compare the SDPR method results with Naive homotopy-like continuation method.
Observe: DSCF($R, N$) is more difficult to solve for larger $R$.

Question: How does $u^*$ behave for increasing $R$?

Compare the SDPR method results with Naive homotopy-like continuation method

1. Choose $R'$, $N$ and a step size $\Delta R$. 
Observe: DSCF($R, N$) is more difficult to solve for larger $R$.

Question: How does $u^*$ behave for increasing $R$?

Compare the SDPR method results with

**Naive homotopy-like continuation method**

1. Choose $R'$, $N$ and a step size $\Delta R$.
2. Solve DSCF(0, $N$) and obtain its unique solution $u^0$. 
Observe: DSCF\((R, N)\) is more difficult to solve for larger \(R\).

Question: How does \(u^*\) behave for increasing \(R\)?

Compare the SDPR method results with Naive homotopy-like **continuation method**

1. Choose \(R', N\) and a step size \(\Delta R\).
2. Solve DSCF\((0, N)\) and obtain its unique solution \(u^0\).
3. Increase \(R^{k-1}: R^k = R^{k-1} + \Delta R\)
Observe: DSCF(R, N) is more difficult to solve for larger R.

Question: How does \( u^* \) behave for increasing \( R \)?

Compare the SDPR method results with Naive homotopy-like continuation method

1. **Choose** \( R', N \) and a step size \( \Delta R \).
2. Solve DSCF(0, N) and obtain its unique solution \( u^0 \).
3. Increase \( R^{k-1} \): \( R^k = R^{k-1} + \Delta R \)
4. Apply Newton’s method to DSCF(\( R^k \), N) starting from \( u^{k-1} \). Obtain solution \( u^k \) as an approximation to a solution of DSCF(\( R^k \), N).
Minimal kinetic energy solution for increasing $R$

Observe: DSCF($R, N$) is more difficult to solve for larger $R$.

Question: How does $u^*$ behave for increasing $R$?

Compare the SDPR method results with Naive homotopy-like continuation method

1. Choose $R'$, $N$ and a step size $\Delta R$.
2. Solve DSCF(0, $N$) and obtain its unique solution $u^0$.
3. Increase $R^{k-1}$: $R^k = R^{k-1} + \Delta R$
4. Apply Newton’s method to DSCF($R^k, N$) starting from $u^{k-1}$.
   Obtain solution $u^k$ as an approximation to a solution of DSCF($R^k, N$).
5. Iterate 3. and 4. until the desired Reynold’s number $R'$ is reached.
Minimal kinetic energy solution for increasing $R$

Observe: DSCF($R$, $N$) is more difficult to solve for larger $R$.

Question: How does $u^*$ behave for increasing $R$?

Compare the SDPR method results with

Naive homotopy-like continuation method

1. Choose $R'$, $N$ and a step size $\Delta R$.
2. Solve DSCF($0$, $N$) and obtain its unique solution $u^0$.
3. Increase $R^{k-1}$: $R^k = R^{k-1} + \Delta R$
4. Apply Newton’s method to DSCF($R^k$, $N$) starting from $u^{k-1}$. Obtain solution $u^k$ as an approximation to a solution of DSCF($R^k$, $N$).
5. Iterate 3. and 4. until the desired Reynold’s number $R'$ is reached.
Observe: DSCF($R, N$) is more difficult to solve for larger $R$.

Question: How does $u^*$ behave for increasing $R$?

Compare the SDPR method results with Naive homotopy-like continuation method

1. Choose $R'$, $N$ and a step size $\Delta R$.
2. Solve DSCF($0, N$) and obtain its unique solution $u^0$.
3. Increase $R^{k-1}$: $R^k = R^{k-1} + \Delta R$
4. Apply Newton's method to DSCF($R^k, N$) starting from $u^{k-1}$. Obtain solution $u^k$ as an approximation to a solution of DSCF($R^k, N$).
5. Iterate 3. and 4. until the desired Reynold's number $R'$ is reached.

Continuation method is not guaranteed to find the minimizer of $F$!
<table>
<thead>
<tr>
<th>$R$</th>
<th>$N_C$</th>
<th>$N_R$</th>
<th>$E_C$</th>
<th>$E_{SDPR(1)}$</th>
<th>$E_{SDPR(2)}$</th>
<th>$E_{min}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.0096</td>
<td>0.0096</td>
<td>0.0096</td>
<td>0.0096</td>
</tr>
<tr>
<td>100</td>
<td>37</td>
<td>13</td>
<td>0.0030</td>
<td>0.0030</td>
<td>0.0030</td>
<td>0.0030</td>
</tr>
<tr>
<td>500</td>
<td>37</td>
<td>13</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
</tr>
<tr>
<td>1000</td>
<td>37</td>
<td>13</td>
<td>5.4e-4</td>
<td>5.0e-4</td>
<td>5.0e-4</td>
<td>5.0e-4</td>
</tr>
<tr>
<td>2000</td>
<td>37</td>
<td>13</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
</tr>
<tr>
<td>4000</td>
<td>37</td>
<td>17</td>
<td>6.3e-4</td>
<td>4.6e-4</td>
<td>4.6e-4</td>
<td>4.6e-4</td>
</tr>
<tr>
<td>6000</td>
<td>36</td>
<td>16</td>
<td>5.7e-4</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
</tr>
<tr>
<td>10000</td>
<td>35</td>
<td>17</td>
<td>4.7e-4</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
</tr>
<tr>
<td>20000</td>
<td>35</td>
<td>17</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>3.3e-4</td>
<td>3.3e-4</td>
</tr>
<tr>
<td>30000</td>
<td>35</td>
<td>17</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>2.5e-4</td>
<td>2.5e-4</td>
</tr>
<tr>
<td>100000</td>
<td>34</td>
<td>16</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>8.8e-5</td>
<td>8.8e-5</td>
</tr>
</tbody>
</table>
CF(R,5)

<table>
<thead>
<tr>
<th>$R$</th>
<th>$N_C$</th>
<th>$N_R$</th>
<th>$E_C$</th>
<th>$E_{SDPR(1)}$</th>
<th>$E_{SDPR(2)}$</th>
<th>$E_{\text{min}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.0096</td>
<td>0.0096</td>
<td>0.0096</td>
<td>0.0096</td>
</tr>
<tr>
<td>100</td>
<td>37</td>
<td>13</td>
<td>0.0030</td>
<td>0.0030</td>
<td>0.0030</td>
<td>0.0030</td>
</tr>
<tr>
<td>500</td>
<td>37</td>
<td>13</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
</tr>
<tr>
<td>1000</td>
<td>37</td>
<td>13</td>
<td>5.4e-4</td>
<td>5e-4</td>
<td>5e-4</td>
<td><strong>5e-4</strong></td>
</tr>
<tr>
<td>2000</td>
<td>37</td>
<td>13</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
<td>6.2e-4</td>
</tr>
<tr>
<td>4000</td>
<td>37</td>
<td>17</td>
<td>6.3e-4</td>
<td>4.6e-4</td>
<td>4.6e-4</td>
<td>4.6e-4</td>
</tr>
<tr>
<td>6000</td>
<td>36</td>
<td>16</td>
<td>5.7e-4</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
</tr>
<tr>
<td>10000</td>
<td>35</td>
<td>17</td>
<td>4.7e-4</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
</tr>
<tr>
<td>20000</td>
<td>35</td>
<td>17</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td><strong>3.3e-4</strong></td>
<td><strong>3.3e-4</strong></td>
</tr>
<tr>
<td>30000</td>
<td>35</td>
<td>17</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>2.5e-4</td>
<td>2.5e-4</td>
</tr>
<tr>
<td>100000</td>
<td>34</td>
<td>16</td>
<td>4.5e-4</td>
<td>4.5e-4</td>
<td>8.8e-5</td>
<td>8.8e-5</td>
</tr>
</tbody>
</table>

SDPR(1) yields the min energy solution for $R \leq 10000$, SDPR(2) yields min energy solution for all $R$. 

Mevisessen (Tokyo Tech) Solutions for Polynomial Systems ISSAC 2009 23 / 26
CF(R,5)
$CF(R,5)$

$u_C(R)$ is not the minimum energy solution!
\( u_C(R) \) is not the minimum energy solution!

\( u^*(R) \) is obtained by SDPR(2) and \( E_{\text{min}}(R) \) tends to 0!
- $u_C(R)$ is not the minimum energy solution!
- $u^*(R)$ is obtained by SDPR(2) and $E_{\text{min}}(R)$ tends to 0!
- Similar behavior can be observed for $N \in \{6, 7\}$. 
Behavior for increasing $R$
Behavior for increasing $R$

**Conjecture**

(a) $F(u_0(N)) = E_{\text{min}}(0, N) \geq E_{\text{min}}(R, N) \geq 0 \quad \forall R \geq 0.$
Behavior for increasing $R$

Conjecture

a) $F(u_0(N)) = E_{\text{min}}(0, N) \geq E_{\text{min}}(R, N) \geq 0 \quad \forall R \geq 0$.

b) $E_{\text{min}}(R, N) \to 0$ for $R \to \infty$. 
Behavior for increasing $R$

Conjecture

a) $F(u_0(N)) = E_{\text{min}}(0, N) \geq E_{\text{min}}(R, N) \geq 0 \quad \forall R \geq 0.$

b) $E_{\text{min}}(R, N) \to 0$ for $R \to \infty.$

a) can be used as a certificate for non-optimality of a solution $u'(R, N)$, if $F(u'(R, N)) > E_{\text{min}}(0, N).$
Behavior for increasing $R$

**Conjecture**

\begin{itemize}
\item[a)] $F(u_0(N)) = E_{\text{min}}(0, N) \geq E_{\text{min}}(R, N) \geq 0 \quad \forall R \geq 0.$
\item[b)] $E_{\text{min}}(R, N) \to 0$ for $R \to \infty.$
\end{itemize}

- a) can be used as a certificate for non-optimality of a solution $u'(R, N)$, if $F(u'(R, N)) > E_{\text{min}}(0, N)$.
- In the case $u_0(N)$ can be continued to $\tilde{u}(R, N)$, $u'(R, N)$ is non-optimal if $F(u'(R, N)) > F(\tilde{u}(R, N))$. 

Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).
Summary

- Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).
- Algorithm for approximately enumerating all solutions of a polynomial system one by one.
Summary

- Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).
- Algorithm for approximately enumerating all solutions of a polynomial system one by one.
- Convergence result.
Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).

Algorithm for approximately enumerating all solutions of a polynomial system one by one.

Convergence result.

Discrete steady cavity flow problem for large $R$ is a challenging test problem for new solvers in numerical algebra.
Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).

Algorithm for approximately enumerating all solutions of a polynomial system one by one.

Convergence result.

Discrete steady cavity flow problem for large $R$ is a challenging test problem for new solvers in numerical algebra.

Future Research
Summary

- Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).
- Algorithm for approximately enumerating all solutions of a polynomial system one by one.
- Convergence result.
- Discrete steady cavity flow problem for large $R$ is a challenging test problem for new solvers in numerical algebra.

Future Research
- Systematic procedure to determine $\epsilon$ and $b$. 
Summary

- Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).
- Algorithm for approximately enumerating all solutions of a polynomial system one by one.
- Convergence result.
- Discrete steady cavity flow problem for large $R$ is a challenging test problem for new solvers in numerical algebra.

Future Research
- Systematic procedure to determine $\epsilon$ and $b$.
- Prove Conjectures.
Summary

- Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).
- Algorithm for approximately enumerating all solutions of a polynomial system one by one.
- Convergence result.
- Discrete steady cavity flow problem for large $R$ is a challenging test problem for new solvers in numerical algebra.

Future Research

- Systematic procedure to determine $\epsilon$ and $b$.
- Prove Conjectures.
- Apply alternative difference scheme to keep physical invariants under discretization.
Summary

- Min energy solution can be found for many $R$ with SDPR(1) or SDPR(2).
- Algorithm for approximately enumerating all solutions of a polynomial system one by one.
- Convergence result.
- Discrete steady cavity flow problem for large $R$ is a challenging test problem for new solvers in numerical algebra.

Future Research

- Systematic procedure to determine $\epsilon$ and $b$.
- Prove Conjectures.
- Apply alternative difference scheme to keep physical invariants under discretization.
- Consider linear instead of quadratic constraints in enumeration algorithm to improve stability.