Three Applications of Semidefinite Programming in 0-1 Quadratic Programming

Relaxation
Reformulation
Reduction

PhD Ray Pörn and MSc Otto Nissfolk

OSE-seminar in Åbo 3.11.2010

- **Background and introduction**
  - Application – ground state for the ***Coulomb glass***
- **0-1 QP (quadratic programming) versus SDP (semidefinite programming)**
- **SDP Relaxation**
  - Lower bounding by SDP
  - Upper bounding heuristic – rank1-truncation and feasibility search
  - Upper bound improvement using randomization
- **Reformulation by SDP**
  - Optimal convexification by diagonal pertubation
- **Reduction using SDP**
  - Combinatorial reduction using optimized integer cuts.
- **Discussion and conclusions**

The Coulomb Glass

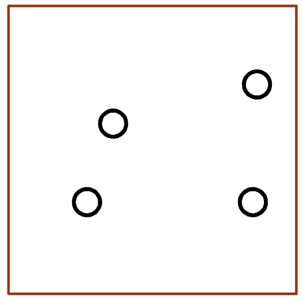
A problem in material physics:

- Appears in practice for **lightly doped semiconductors** at low temperature
- At low temperature the movement of the charges are slow and the energy of the configuration decreases slowly towards the minimal energy configuration
- This slow relaxation towards lower and lower energies is called a **Coulomb glass**
- The objective is to find the **ground state** of a material, i.e. the configuration of charges with minimal energy.
- From the ground state it is possible to compute the so called **Coulomb gap** of the material (observed experimentally)
- The Coulomb gap gives information about **conductivity**

The Coulomb Glass

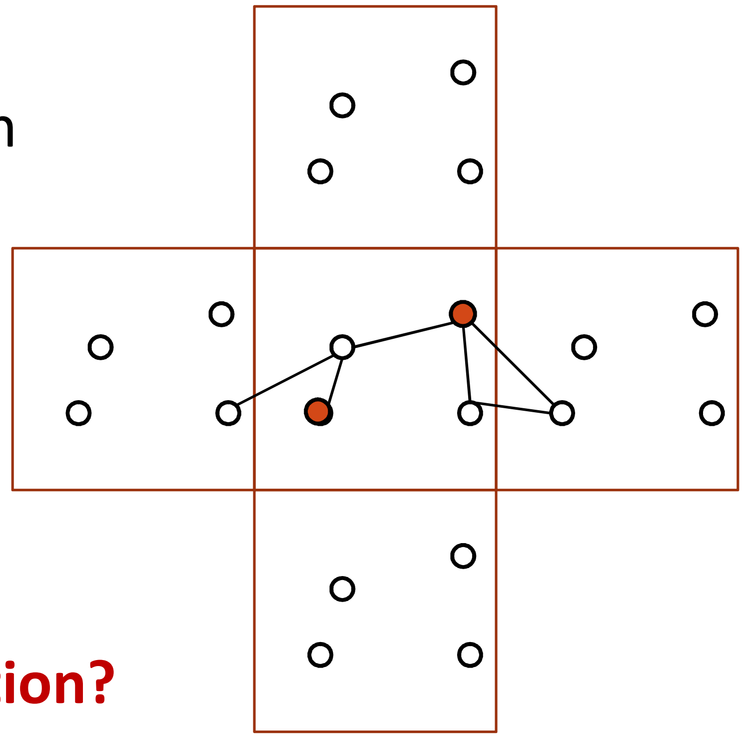
Problem description

n sites in space



Periodical boundary constraints

Shortest distances between all sites – r_{ij}



$n/2$ sites are filled with charges (electrons)

Minimal energy configuration?

THE COULOMB GLASS

- Minimize the total energy of the system when exactly half of the sites are filled

  Energy = Coulomb interaction + site specific energy

Coulomb interaction between electrons at sites *i* and *j*: $k_e \frac{q_e^2}{r_{ij}}$

Site energy at site *i*: ε_i

Binary variables:

x_i = 0 if site *i* is empty

x_i = 1 if site *i* is occupied

$\Rightarrow$ $k_e \frac{q_e^2 x_i x_j}{r_{ij}}$ $\varepsilon_i x_i$

Sum over all possible combinations: $E(x) = k_e q_e^2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{x_i x_j}{r_{ij}} + \sum_{i=1}^{n} \varepsilon_i x_i$

The Coulomb Glass

The Coulomb glass can be posed as a **linearly equality constrained 0-1 quadratic programming problem**

$$\min_x \quad E(x) = k_e q_e^2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{x_i x_j}{r_{ij}} + \sum_{i=1}^{n} \varepsilon_i x_i$$

$$\sum_{i=1}^{n} x_i = \frac{n}{2}, \quad x \in \{0,1\}^n$$

normalize ➔ $$\min_x \quad E(x) = x^T Q x + c^T x$$

$$e^T x = \frac{n}{2}, \quad x \in \{0,1\}^n$$

where $Q_{ij} = \begin{cases} 1/(2r_{ij}), & i \neq j \\ 0, & i = j \end{cases}$, $c_i = \varepsilon_i$ and $e = (1,1,\ldots,1)^T$

0-1 QP VERSUS SDP

- Consider a 0-1 QP with linear constraints

(QP)
$$\min_x \; x^T Q x + c^T x$$
$$Ax = a, \quad Bx \le b$$
$$x \in \{0,1\}^n$$

nonconvexities

- Equivalent SDP formulation

(SDP)
$$\min_{X \in S^n, x \in R^n} \; Q \bullet X + c^T x$$
$$Ax = a, \quad Bx \le b$$
$$\operatorname{diag}(X) = x, \quad X = xx^T \qquad (\operatorname{rank}(X) = 1)$$

where $Q \bullet X = \langle Q, X \rangle = \operatorname{trace}(QX) = \sum_{i=1}^{n} \sum_{j=1}^{n} Q_{ij} X_{ij}$

0-1 QP VERSUS SDP

- The rank-1 constraints are relaxed

$$X - xx^T = 0 \mapsto \quad X - xx^T \succeq 0 \quad \Leftrightarrow \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0$$

and the Semidefinite relaxation of QP is obtained (Goemans and Williamson 1995)

(SDR)
$$\begin{aligned} \min_{X \in S^n, x \in R^n} \quad & Q \bullet X + c^T x \\ & Ax = a, \quad Bx \le b \\ & \text{diag}(X) = x, \quad \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0 \end{aligned}$$

- Include squared-norm constraints to strengthen the relaxation

$$Ax = a \Leftrightarrow \|Ax - a\|^2 = 0 \Leftrightarrow x^T A^T Ax - 2a^T Ax + \|a\|^2 = 0$$
$$\Leftrightarrow (A^T A) \bullet X - 2a^T Ax + \|a\|^2 = 0$$

Relaxation of the Coulomb Glass

(CG-QP) $\min_{x\in\{0,1\}^n} \quad x^T Qx + c^T x$

$$e^T x = \frac{n}{2}$$

$\min_{X\in S^n, x\in R^n} \quad Q \bullet X + c^T x$ (CG-SDR)

$$e^T x = \frac{n}{2}, \qquad (ee^T)\bullet X - ne^T x + \frac{n^2}{4} = 0$$

$$\mathrm{diag}(X) = x, \qquad \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0$$

- Possible to describe with {-1,1} variables instead of {0,1}.

  ➔ translation y=2x-1 ➔ More homogeneous SDP

$$e^T x = \frac{n}{2} \quad \Leftrightarrow \quad e^T y = 0$$

(CG-hSDR) $\min_{Y\in S^{n+1}} \quad \hat{Q}\bullet Y$

$$\hat{P}\bullet Y = 0$$

$$\mathrm{diag}\,(Y) = 1, \qquad Y \succeq 0$$

$$\hat{Q} = \frac{1}{4}\begin{bmatrix} Q & c+Qe \\ (c+Qe)^T & 0 \end{bmatrix}$$

$$\hat{P} = \frac{1}{2}\begin{bmatrix} 0 & e \\ e^T & 0 \end{bmatrix}$$

COMPUTATIONAL EXPERIMENTS WITH HSDR

Lower bounding of QP by SDP

SDR is a relaxation of QP ➔ v(CG-SDR) ≤ v(CG-QP)

mean CPU-time (s)	0,6
iterations	15
variables	1326
constraints	52

N	Instance	LB
50	1	111,015
50	2	114,762
50	3	116,734
50	4	115,937
50	5	114,636
50	6	115,328
50	7	114,862
50	8	121,055
50	9	116,956
50	10	112,298

mean CPU-time (s)	1,5
iterations	18
variables	5151
constraints	102

N	Instance	LB
100	1	362,186
100	2	363,163
100	3	359,391
100	4	363,895
100	5	370,075
100	6	364,915
100	7	363,182
100	8	362,282
100	9	355,044
100	10	364,853

mean CPU-time (s)	3,6
iterations	20
variables	11476
constraints	152

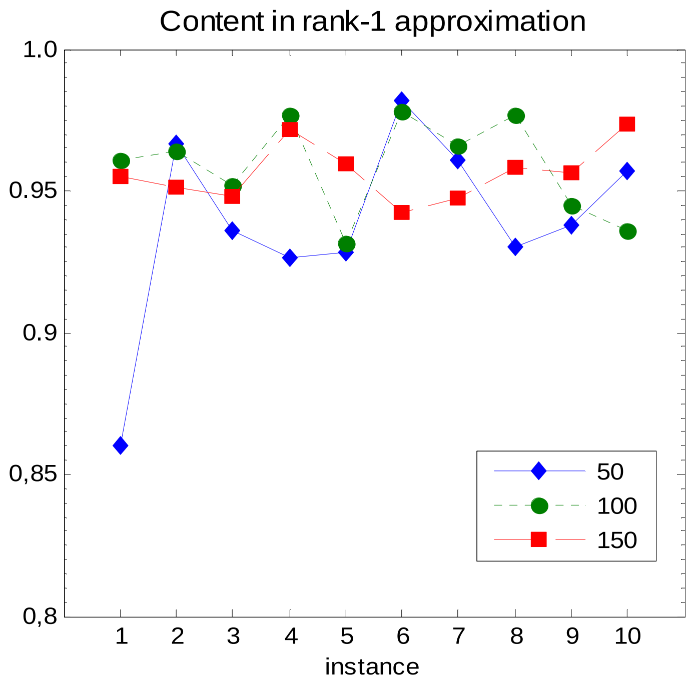
N	Instance	LB
150	1	698,396
150	2	692,320
150	3	691,742
150	4	692,439
150	5	694,263
150	6	689,303
150	7	689,102
150	8	682,926
150	9	678,391
150	10	683,102

Optimizer: Sedumi using cvx in matlab 2007b

Computer: Dell Latitude Laptop, Intel Core 2 Duo P8400, 2,26 GHz, 3,45 GB of RAM

Computational experiments with hSDR

○ Analyzing the SDR solution



Rank is almost constant with problem size.

About 95% of Y is explained in the rank-1 solution.

COMPUTATIONAL EXPERIMENTS WITH HSDR

- **Upper bounding by rank-1 truncation and feasibility search**

$$Y_{sdp} = \sum_{i=1}^{r} \lambda_i q_i q_i^T \Rightarrow Y_{r1} = \lambda_1 q_1 q_1^T \Rightarrow y_{r1} = \sqrt{\lambda_1} q_1$$

Algorithm: Feasibility Search (FS)

Given an approximate solution y_{approx}

y=sign(y_{approx})

while (sum(y)<0)

    for each y_i with y_i==-1

        z = y+2e_i

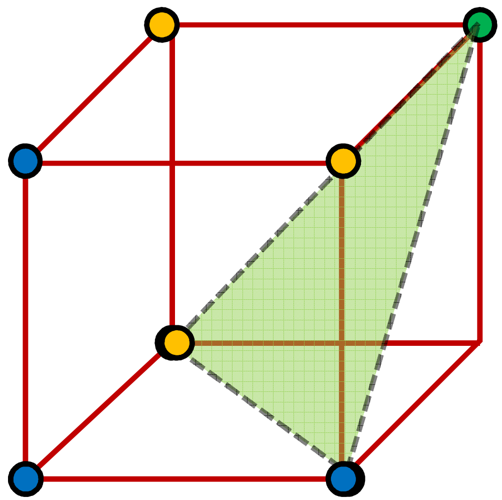
        df(z,y) = E(z)-E(y)

    end

    zbest = argmin $_z$ {dE(z,y)}

    y = zbest

end



COMPUTATIONAL EXPERIMENTS WITH HSDR

- **Upper bounding by randomization and feasibility search.**

The SDR solution *Y** can be considered as a covariance matrix since it is symmetric and positive semidefinite. It is also usually of very low rank.

Randomization

Given a solution Y* and a number of randomizations ITER

for i = 1,…,ITER

generate $yrand_i \sim N(0,Y^*)$ and apply FS to obtain a feasible point $yfeas_i$

end

determine $i^* = \mathrm{argmin}_i \; \{ f(yfeas_i) \}$

$yfeas_{i*}$ is an upper bound (approximate solution) to QP

174.6

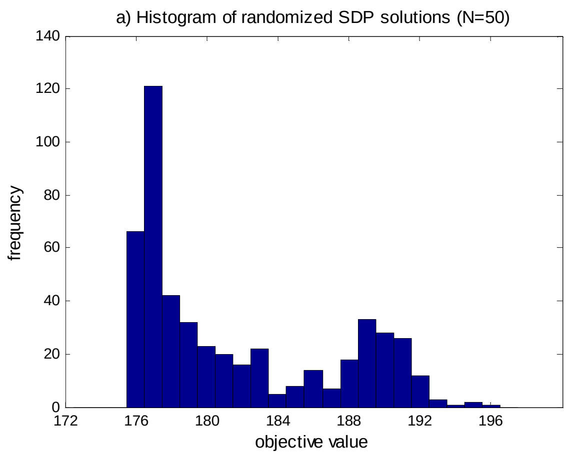
COMPUTATIONAL EXPERIMENTS WITH HSDR

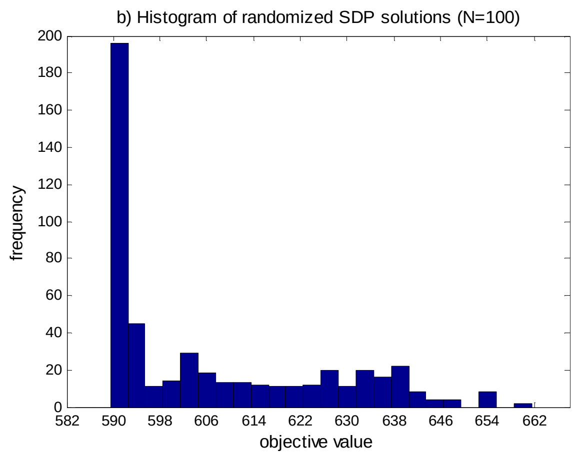
○ Upper bounding by randomization (ITER=500)

LB =174.6

UB_{r1} =176.6

UB_{rand} =175.9





LB=586.3

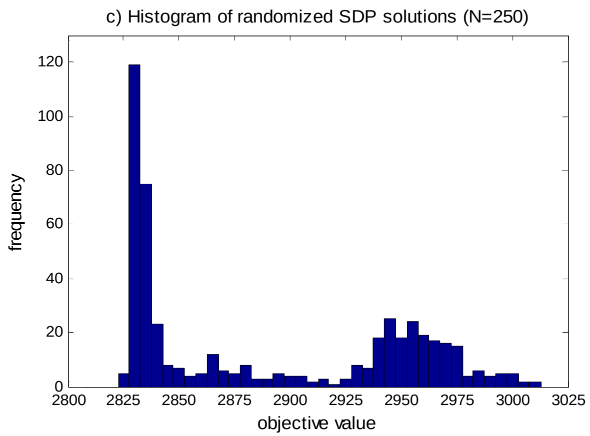
UB_{r1} =590.8

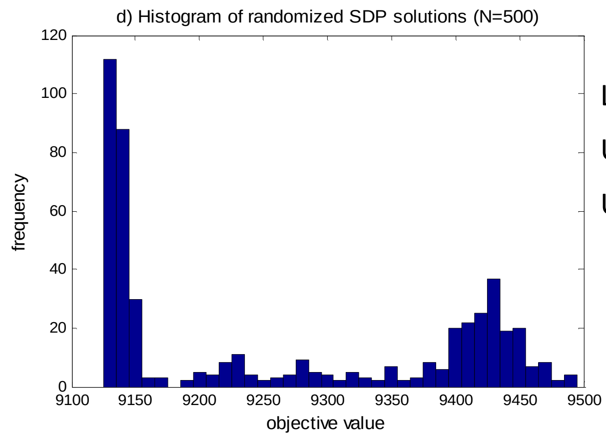
UB_{rand} =589.8

LB=2819.2

UB_{r1}=2827.1

UB_{rand}=2827.3





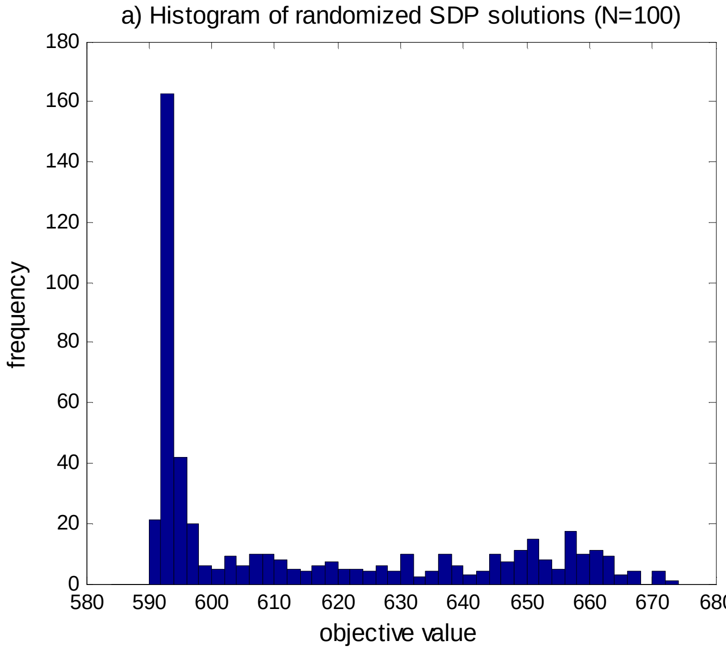
LB=9107.6

UB_{r1}=9128.4

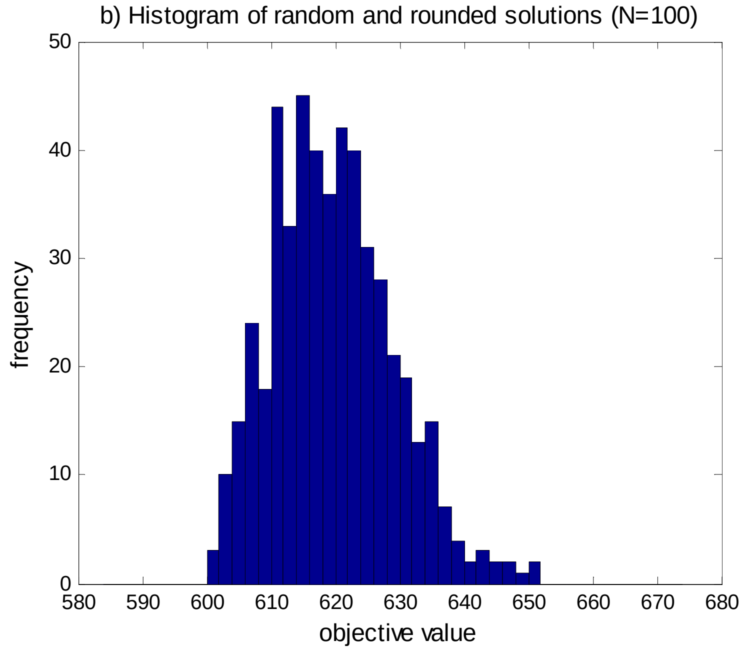
UB_{rand}=9126.4

COMPUTATIONAL EXPERIMENTS WITH HSDR

- Not all randomizations are good . . . .



y ~ N(0,Y*) + FS



y ~ sign(U(-1,1)) + FS

- The space contains good solutions but they are NOT easily found!

Computational experiments with hSDR

Lower bounding and upper bounding heuristics

2D Coulomb glass, N=50

mean CPU-time (s)	0,6
iterations	19
variables	1326
constraints	52

N	Instance	LB	UB-rank1	UB-rand	Improvement	UB-LB	gap %
50	1	111,015	119,384	**117,410**	1,97	6,39	5,45 %
50	2	114,762	117,402	**116,483**	0,92	1,72	1,48 %
50	3	116,734	121,229	**121,064**	0,17	4,33	3,58 %
50	4	115,937	120,272	**117,346**	2,93	1,41	1,20 %
50	5	114,636	118,964	**118,856**	0,11	4,22	3,55 %
50	6	115,328	120,934	**120,648**	0,29	5,32	4,41 %
50	7	114,862	117,553	**116,974**	0,58	2,11	1,81 %
50	8	121,055	122,736	**122,473**	0,26	1,42	1,16 %
50	9	116,956	121,294	**120,860**	0,43	3,90	3,23 %
50	10	112,298	**116,310**	**116,310**	0,00	4,01	3,45 %

COMPUTATIONAL EXPERIMENTS WITH HSDR

- **Lower bounding and upper bounding heuristics**

  2D Coulomb glass, N=100

mean CPU-time (s)	1,5
iterations	20
variables	5151
constraints	102

N	Instance	LB	UB-rank1	UB-rand	Improvement	UB-LB	gap %
100	1	362,186	369,323	**369,067**	0,26	6,88	1,86 %
100	2	363,163	370,354	**369,853**	0,50	6,69	1,81 %
100	3	359,391	367,364	**367,150**	0,21	7,76	2,11 %
100	4	363,895	368,235	**367,847**	0,39	3,95	1,07 %
100	5	370,075	374,178	**373,205**	0,97	3,13	0,84 %
100	6	364,915	371,351	**370,330**	1,02	5,42	1,46 %
100	7	363,182	**366,032**	**366,032**	0,00	2,85	0,78 %
100	8	362,282	366,343	**366,282**	0,06	4,00	1,09 %
100	9	355,044	366,362	**365,775**	0,59	10,73	2,93 %
100	10	364,853	371,466	**369,878**	1,59	5,03	1,36 %

COMPUTATIONAL EXPERIMENTS WITH HSDR

Lower bounding and upper bounding heuristics

2D Coulomb glass, N=150

Mean CPU-time (s)	3,6
iterations	23
variables	11476
constraints	152

N	Instance	LB	UB-rank1	UB-rand	Improvement	UB-LB	gap %
150	1	698,396	707,786	**706,740**	1,05	8,34	1,18 %
150	2	692,320	701,661	**699,132**	2,53	6,81	0,97 %
150	3	691,742	703,112	**702,781**	0,33	11,04	1,57 %
150	4	692,439	694,816	**694,594**	0,22	2,16	0,31 %
150	5	694,263	702,746	**699,080**	3,67	4,82	0,69 %
150	6	689,303	697,074	**696,653**	0,42	7,35	1,06 %
150	7	689,102	702,491	**701,978**	0,51	12,88	1,83 %
150	8	682,926	697,260	**696,293**	0,97	13,37	1,92 %
150	9	678,391	686,707	**683,890**	2,82	5,50	0,80 %
150	10	683,102	693,917	**689,111**	4,81	6,01	0,87 %

REFORMULATION BY SDP

- SDP can be used as a **reformulation** tool.
- QP can be convexified in an optimal way.
- Pertubation of *Q* by a **constant diagonal matrix**.

  ➔ Minimal eigenvalue reformulation

$$x^T Qx + c^T x = x^T \underbrace{(Q - \lambda_{\min} I)}_{PSD} x + (c + \lambda_{\min} e)^T x$$

- Pertubation by an **arbitrary diagonal matrix** Diag(*u*).

$$x^T Qx + c^T x = x^T \underbrace{(Q + \mathrm{Diag}(u))}_{PSD} x + (c - u)^T x$$

➔ How can we find a vector *u* that makes *Q*+Diag(*u*) PSD and produces a tight relaxation of the perturbed QP?

Reformulation by SDP

- Formally, we wish to solve

$$\max_{\substack{u \in R^n \\ Q+\text{Diag}(u) \succeq 0}} \quad \min_{x \in R^n} x^T (Q + \text{Diag}(u))x + (c-u)^T x$$

- According to Billionett et. al. 2007 the optimal u is the solution to

$$\max_{u \in R^n} \quad -\frac{1}{4}^{T} (c-u)^T (Q + \text{Diag}(u))^+ (c-u)$$

$$Q + \text{Diag}(u) \succeq 0$$

or

$$\max_{r \in R, u \in R^n} \quad r$$

$$r \leq -\frac{1}{4}^{T} (c-u)^T (Q + \text{Diag}(u))^+ (c-u)$$

$$Q + \text{Diag}(u) \succeq 0$$

This problem has a **quadratic constraint** and a **semidefinite restriction** on a matrix.

REFORMULATION BY SDP

- Using Schur's lemma the quadratic and semidefinite constraint can jointly be written as

$$\max_{r \in R, u \in R^n} \quad r$$

$$\begin{bmatrix} -r & \frac{1}{2}(c-u)^T \\ \frac{1}{2}(c-u) & Q + \mathrm{Diag}(u) \end{bmatrix} \succeq 0$$

The **dual** of this SDP problem is **exactly** the semidefinite relaxation of QP (if we omit the constraints)

$$\min_{X \in S^n, x \in R^n} \quad Q \bullet X + c^T x$$

$$\mathrm{diag}(X) = x$$

$$\begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0$$

The optimal *u* vector is the Lagrange multiplicators of the constraint diag(*X*)=*x*

Reformulation by SDP

- 30 instances of the 2D Coulomb glass with sizes *n*=50, 100, 150.

- Global optimal solutions for convex QP by optimal u-pertubation. Solves *n*=150 problems to global optimality in less than 10 minutes.

- *n*=50 instance 1: >1hour for minimal eigenvalue reformulation (typical time ~ 1 min)

Optimizer: CPLEX 12.2.0.0

Computer: Intel Core i7 930 PC, 2,8 GHz, 6 GB

N	Instance	fopt	froot	time (s)	nodes
50	1	116,986	111,015	0,28	1369
50	2	116,483	114,762	0,23	273
50	3	120,925	116,734	0,22	389
50	4	117,291	115,937	0,23	1238
50	5	118,702	114,636	0,22	1221
50	6	120,613	115,328	0,27	801
50	7	116,974	114,862	0,28	576
50	8	122,411	121,055	0,17	393
50	9	120,781	116,956	0,20	573
50	10	116,310	112,298	0,20	630
100	1	368,878	362,186	1,16	9286
100	2	368,924	363,163	0,86	5600
100	3	366,702	359,391	8,47	120220
100	4	367,242	363,895	0,87	5841
100	5	372,843	370,075	6,62	97779
100	6	370,044	364,915	1,12	8550
100	7	366,032	363,182	0,83	4966
100	8	366,244	362,282	0,83	5441
100	9	365,730	355,044	2,06	22966
100	10	369,736	364,853	14,32	221894
150	1	706,279	698,396	251,65	1621920
150	2	698,707	692,320	206,00	1108800
150	3	701,798	691,742	521,04	2824610
150	4	694,545	692,439	2,28	8686
150	5	698,331	694,263	65,16	475828
150	6	695,910	689,303	149,06	802038
150	7	701,669	689,102	443,03	2830250
150	8	695,642	682,926	60,45	353152
150	9	683,801	678,391	128,98	881831
150	10	688,794	683,102	36,10	234828

REFORMULATION BY SDP

N	Instance	LB	UB-rank1	UB-rand	opt	gap to opt %
50	1	111,015	119,384	117,410	116,986	**0,36 %**
50	2	114,762	117,402	**116,483**	**116,483**	**0,00 %**
50	3	116,734	121,229	121,064	120,925	**0,11 %**
50	4	115,937	120,272	117,346	117,291	**0,05 %**
50	5	114,636	118,964	118,856	118,702	**0,13 %**
50	6	115,328	120,934	120,648	120,613	**0,03 %**
50	7	114,862	117,553	**116,974**	**116,974**	**0,00 %**
50	8	121,055	122,736	122,473	122,411	**0,05 %**
50	9	116,956	121,294	120,860	120,781	**0,07 %**
50	10	112,298	**116,310**	**116,310**	**116,310**	**0,00 %**

N	Instance	LB	UB-rank1	UB-rand	opt	gap to opt %
100	1	362,186	369,323	369,067	368,878	**0,05 %**
100	2	363,163	370,354	369,853	368,924	**0,25 %**
100	3	359,391	367,364	367,150	366,702	**0,12 %**
100	4	363,895	368,235	367,847	367,242	**0,16 %**
100	5	370,075	374,178	373,205	372,843	**0,10 %**
100	6	364,915	371,351	370,330	370,044	**0,08 %**
100	7	363,182	**366,032**	**366,032**	**366,032**	**0,00 %**
100	8	362,282	366,343	366,282	366,244	**0,01 %**
100	9	355,044	366,362	365,775	365,730	**0,01 %**
100	10	364,853	371,466	369,878	369,736	**0,04 %**

The randomization heuristic gives us consistently optimal or very near optimal solutions.

➔ Possible approach for large scale problems (*n*~1000)

N	Instance	LB	UB-rank1	UB-rand	opt	gap to opt %
150	1	698,396	707,786	706,740	706,279	**0,07 %**
150	2	692,320	701,661	699,132	698,707	**0,06 %**
150	3	691,742	703,112	702,781	701,798	**0,14 %**
150	4	692,439	694,816	694,594	694,545	**0,01 %**
150	5	694,263	702,746	699,080	698,331	**0,11 %**
150	6	689,303	697,074	696,653	695,910	**0,11 %**
150	7	689,102	702,491	701,978	701,669	**0,04 %**
150	8	682,926	697,260	696,293	695,642	**0,09 %**
150	9	678,391	686,707	683,890	683,801	**0,01 %**
150	10	683,102	693,917	689,111	688,794	**0,05 %**

COMBINATORIAL REDUCTION USING SDP

- Given a good candidate solution *xc* to (CG-QP).

  Idea: How large ball centered at *xc* is needed to ensure that the global optimum is contained in the ball?

  Find smallest *r*-value so that the constraint $\|x - xc\|_1 \leq r$ can be added to CG-QP without affecting the global optimum.

When is

$$\min_{x \in \{0,1\}^n} x^T Q x + c^T x \quad e^T x = \frac{n}{2} \qquad \Leftrightarrow \qquad \min_{x \in \{0,1\}^n} x^T Q x + c^T x \quad e^T x = \frac{n}{2}, \quad \|x - xc\|_1 \leq r$$

For small *r*-values the problem to the right will be "easy".

Note: $\|x - xc\|_1 = \sum_{i=1}^{n} |x_i - xc_i| = \sum_{i \in B^0} x_i - \sum_{i \in B^1} x_i + \underbrace{|B^1|}_{n/2} \leq r$

COMBINATORIAL REDUCTION USING SDP

- Given a good candidate solution *xc*. Consider the problem

$$\max_{x\in\{0,1\}^n, r} \quad r$$

$$x^T Q x + c^T x = E(xc)$$

$$e^T x = \frac{n}{2}, \quad \|x - xc\|_1 \geq r$$

with SDP relaxation

$$\max_{X\in S^n, x\in R^n, r\in R} \quad r$$

$$Q \bullet X + c^T x = E(xc)$$

$$e^T x = \frac{n}{2}, \quad \|x - xc\|_1 \geq r$$

$$\operatorname{diag}(X) = x, \quad \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0, \quad X \geq 0$$

If r^* is optimal in problem above then we can include $\|x - xc\|_1 \leq r^*$ in problem (CG-QP) without affecting the global optimum.

Combinatorial Reduction Using SDP

Some basic models of (CG-QP)

- Linear model 1:

  Standard linearization

  O(n^2) additional vars

$$\min_{x_i \in \{0,1\}, z_{ij} \in [0,1]} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} Q z_{ij} + \sum_{i=1}^{n} (c_i + Q_{ii}) x_i$$

$$\sum_{i=1}^{n} x_i = \frac{n}{2}, \quad z_{ij} \geq x_i + x_j - 1 \qquad \forall i \neq j$$

- Linear model 2:

  Compact linearization

  O(n) additional vars

$$\min_{x_i \in \{0,1\}, y_i, w_i \in R} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} Q w_{ij} + \sum_{i=1}^{n} c_i x_i$$

$$\sum_{i=1}^{n} x_i = \frac{n}{2}, \quad y_i = \sum_{j=1}^{n} Q_{ij} x_j \quad \forall i$$

$$w_i \geq y_i - y_{i,\max}(1 - x_i), \quad w_i \geq 0$$

$$y_{i,\min} \leq y_i \leq y_{i,\max}$$

- Simple quadratic model:

  Diagonal pert. by smallest eigenvalue

  No additional vars

$$\min_{x \in \{0,1\}^n} \quad x^T (Q - \lambda_{\min} I) x + (c + \lambda_{\min} e)^T x$$

$$e^T x = \frac{n}{2}$$

Combinatorial Reduction Using SDP

Candidate solution *xc* is obtained from SDR randomization procedure with (ITER=200).

		Model 1					*Model 1 with norm constraint*						
N	**i**	**UB**	**LB**	**fopt**	**time(s)**	**nodes**	**UB**	**LB**	**fopt**	**time(s)**	**nodes**	**r**	**Improvement**
50	1	117,57	102,45	116,99	3600	40597	117,21	102,22	116,99	3600	45769	18	**0,00 %**
50	2	116,53	105,18	116,48	3600	38732	116,48	OPT	116,48	793	63861	6	**77,98 %**
50	3	121,42	112,48	120,92	3600	37880	120,92	OPT	120,92	440	41680	6	**87,77 %**
50	4	117,39	103,07	117,29	3600	33865	117,29	112,72	117,29	3600	95510	10	**0,00 %**
50	5	118,87	106,06	118,70	3600	38140	118,75	114,71	118,70	3600	150281	12	**0,00 %**
50	6	120,81	107,39	120,61	3600	37976	120,61	OPT	120,61	1405	158387	6	**60,97 %**
50	7	117,51	105,31	116,97	3600	39230	116,97	OPT	116,97	675	53903	6	**81,26 %**
50	8	122,95	112,54	122,41	3600	54835	122,41	OPT	122,41	636	45147	6	**82,34 %**
50	9	121,25	108,30	120,78	3600	36743	120,78	OPT	120,78	2729	169346	8	**24,18 %**
50	10	117,35	102,38	116,31	3600	34273	116,31	OPT	116,31	8	5565	4	**99,77 %**

		Model 2					*Model 2 with norm constraint*						
N	**i**	**UB**	**LB**	**fopt**	**time(s)**	**nodes**	**UB**	**LB**	**fopt**	**time(s)**	**nodes**	**r**	**Improvement**
50	1	116,99	109,75	116,99	3600	60902698	116,99	111,30	116,99	3600	46763306	18	**0,00 %**
50	2	116,48	112,86	116,48	3600	47401951	116,48	OPT	116,48	3,9	83242	6	**99,89 %**
50	3	120,92	120,91	120,92	3600	44205082	120,92	OPT	120,92	3,0	55791	6	**99,92 %**
50	4	117,29	113,67	117,29	3600	46796452	117,29	OPT	117,29	143	3462266	10	**96,02 %**
50	5	118,70	114,16	118,70	3600	45459945	118,70	OPT	118,70	1339	21346874	12	**62,79 %**
50	6	120,61	115,09	120,61	3600	47518866	120,61	OPT	120,61	4,6	108031	6	**99,87 %**
50	7	116,97	115,93	116,97	3600	42973364	116,97	OPT	116,97	3,5	69565	6	**99,90 %**
50	8	122,41	OPT	122,41	2953	37921734	122,41	OPT	122,41	3,3	73518	6	**99,89 %**
50	9	120,78	118,73	120,78	3600	42162305	120,78	OPT	120,78	42	1240267	8	**98,82 %**
50	10	116,31	112,74	116,31	3600	43060351	116,31	OPT	116,31	0,9	9503	4	**99,97 %**

COMBINATORIAL REDUCTION USING SDP

		Model 3					*Model 3 with norm constraint*							
N	i	UB	LB	fopt	time	nodes	UB	LB	fopt	time	nodes	r	Improvement	-lambda
50	1	116,99	OPT	116,99	987	30745821	116,99	OPT	116,99	395	14976906	18	**59,93 %**	20
50	2	116,48	OPT	116,48	8,5	387956	116,48	OPT	116,48	0,5	12270	6	**94,02 %**	11
50	3	120,92	OPT	120,92	12,5	570151	120,92	OPT	120,92	0,6	14081	6	**95,59 %**	25
50	4	117,29	OPT	117,29	4,6	221310	117,29	OPT	117,29	1,6	65269	10	**65,00 %**	9
50	5	118,70	OPT	118,70	72	3159788	118,70	OPT	118,70	25	1173159	12	**65,37 %**	17
50	6	120,61	OPT	120,61	64	2521113	120,61	OPT	120,61	0,8	28302	6	**98,74 %**	21
50	7	116,97	OPT	116,97	2,5	106409	116,97	OPT	116,97	0,4	7002	6	**84,08 %**	10
50	8	122,41	OPT	122,41	4,1	373514	122,41	OPT	122,41	0,5	11376	6	**88,38 %**	8
50	9	120,78	OPT	120,78	98	4125920	120,78	OPT	120,78	4,4	213867	8	**95,52 %**	20
50	10	116,31	115,56	116,31	3600	98312602	116,31	OPT	116,31	0,3	3672	4	**99,99 %**	32

- The norm constraint is often VERY effective in reducing the combinatorial space.
- Drawback: SDP problem requires elementwise bounds $X \geq 0$ to be effective ➔ Problems with n=200 takes about 30 minutes to solve and for n=300 the solution takes several hours.

CONCLUSIONS

- A 0-1 QP model was derived for the Coulomb glass.
- Tight lower bounding by SDP relaxation.
- Fast upper bounding by randomization and feasibility search.
- Optimal convexification by the QCR-method.
- Global optimal solutions up to about n=150-200.
- “Fast” heuristics by SDR up to n=1500.
- Efficient combinatorial reduction for 0-1 QPs by SDP.

THANK YOU!

… only low rank questions allowed …