Memory size in the Prisoner's Dilemma

Nikoleta E. Glynatsi

Supervised by:

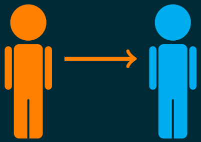
Dr. Vincent Knight   Dr. Jonathan Gillard

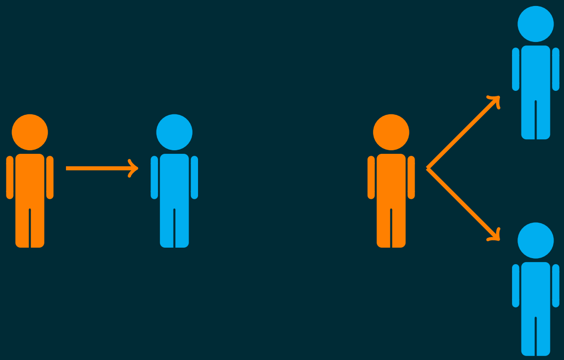
	C	D
C	3 3	5 0
D	0 5	1 1

$$\begin{array}{lcccccc} & 1 & 2 & 3 & \cdots & \overbrace{n-1}^{\text{memory 1}} & n \\ \text{player } p & C & C & C & \cdots & D & \cdots \\ \text{player } q & C & C & D & \cdots & D & \cdots \end{array}$$

memory size $n-1$

William H. Press and Freeman J. Dyson. Iterated Prisoner's Dilemma contains strategies that dominate any evolutionary opponent. 2012



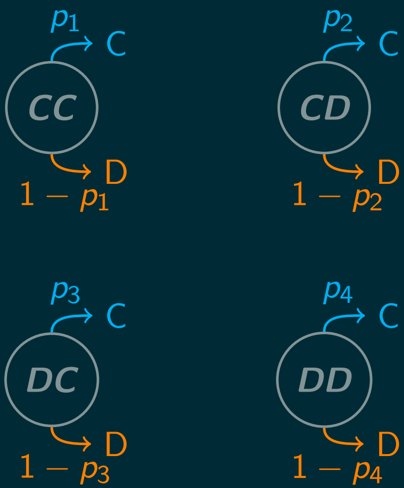


WHICH IS THE BEST MEMORY ONE STRATEGY?

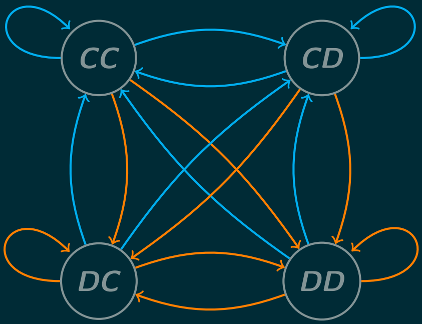
ARE THERE LIMITATIONS TO MEMORY ONE STRATEGIES?

WHICH IS THE BEST MEMORY ONE STRATEGY?

ARE THERE LIMITATIONS TO MEMORY ONE STRATEGIES?



$$\boldsymbol{p} = (p_1, p_2, p_3, p_4) \in \mathbb{R}^4_{[0,1]}$$



$$\begin{bmatrix} p_1 q_1 & p_1\left(-q_1+1\right) & q_1\left(-p_1+1\right) & \left(-p_1+1\right)\left(-q_1+1\right) \\ p_2 q_3 & p_2\left(-q_3+1\right) & q_3\left(-p_2+1\right) & \left(-p_2+1\right)\left(-q_3+1\right) \\ p_3 q_2 & p_3\left(-q_2+1\right) & q_2\left(-p_3+1\right) & \left(-p_3+1\right)\left(-q_2+1\right) \\ p_4 q_4 & p_4\left(-q_4+1\right) & q_4\left(-p_4+1\right) & \left(-p_4+1\right)\left(-q_4+1\right) \end{bmatrix}$$

$$\mathbf{max}_{\boldsymbol{p}}\, \boldsymbol{u_q}(\boldsymbol{p}) \text{ such that } \boldsymbol{p} \in \mathbb{R}^{\mathbf{4}}_{\mathbf{[0,1]}}$$

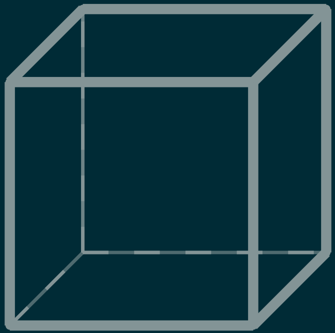
Lemma

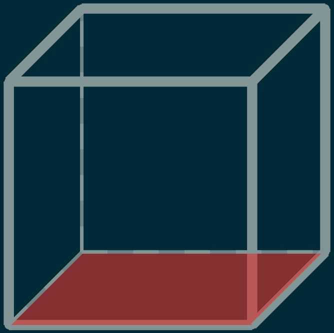
$$u_q(p) = \frac{\frac{1}{2}pQp^T + c^T p + a}{\frac{1}{2}p\bar{Q}p^T + \bar{c}^T p + \bar{a}}$$

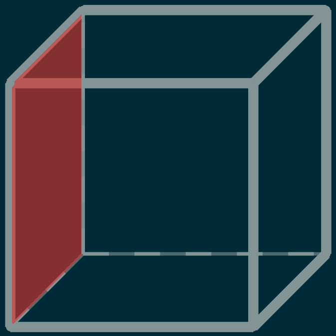
- $Q, \bar{Q} \in \mathbb{R}^{4\times 4}$
- $c, \bar{c} \in \mathbb{R}^{4\times 1}$
- $a, \bar{a} \in \mathbb{R}$

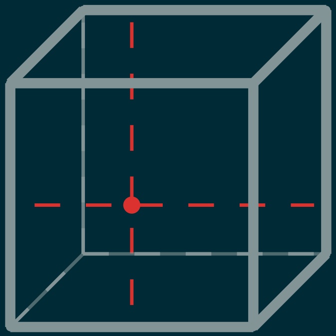
Lemma

$$\frac{1}{N}\sum_{i=1}^{N} u_q{}^{(i)}(p) = \frac{1}{N}\frac{\sum_{i=1}^{N}\left(\frac{1}{2}pQ^{(i)}p^T + c^{(i)T}p + a^{(i)}\right)\prod_{\substack{j=1\\ j\neq i}}^{N}\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)T}p + \bar{a}^{(i)}\right)}{\prod_{i=1}^{N}\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)T}p + \bar{a}^{(i)}\right)}$$







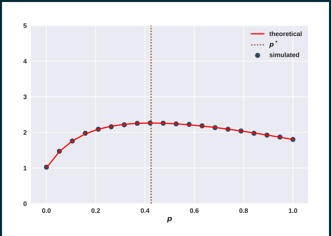


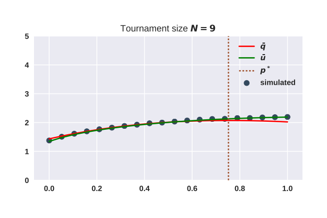
PURELY RANDOM

$$p = (p, p, p, p)$$

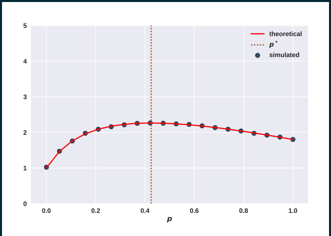
$$\mathbf{S_q} = \mathbf{U}_{i=1}^{\mathbf{2N}} \lambda_{\mathbf{i}} \cup \{\mathbf{0}, \mathbf{1}\}$$

$$\mathbf{1} \leq |\mathbf{S_{q(i)}}| \leq \mathbf{2N} + \mathbf{2}$$

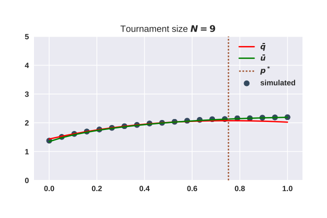




$$\mathbf{S_q} = \mathbf{U}_{i=1}^{\mathbf{2N}} \lambda_{\mathbf{i}} \cup \{\mathbf{0}, \mathbf{1}\}$$



$$\mathbf{1} \leq |\mathbf{S_{q(i)}}| \leq \mathbf{2N} + \mathbf{2}$$



result: optimal behaviour using eigenvalues of companion matrix

REACTIVE

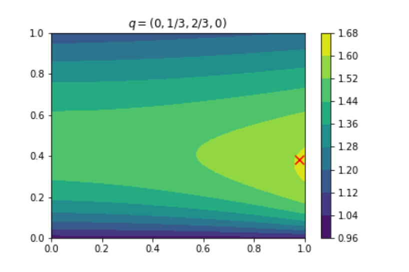
$$p = (p_1, p_2, p_1, p_2)$$

$$\frac{du}{dp_1} = 0$$

$$M = \begin{bmatrix} \cdots \\ \cdots \\ \cdots \end{bmatrix}$$

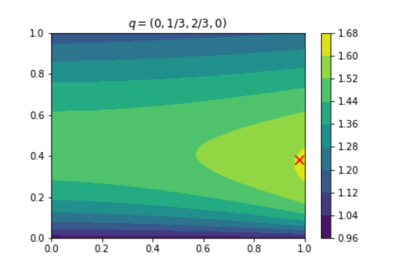
$$\det(M) = 0$$

$$\frac{du}{dp_2} = 0$$



$$\frac{du}{dp_1} = 0 \qquad M = \begin{bmatrix} \cdots \\ \cdots \\ \cdots \end{bmatrix}$$

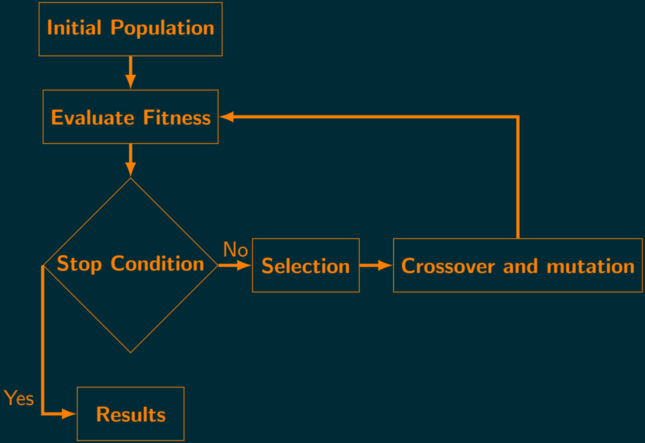
$$\frac{du}{dp_2} = 0 \qquad \det(M) = 0$$



result: optimal behaviour using Sylvester's resultant (Sylvester 1840)

MEMORY ONE

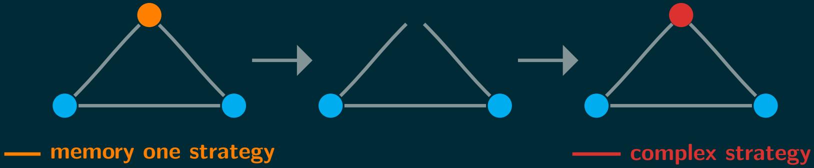
$$\mathbf{b}' = \mathbf{b_0} + \mathbf{m} \times (\mathbf{p}^{(\mathbf{i})} - \mathbf{p}^{(\mathbf{j})})$$



	q_1	q_2	q_3	q_4	p_1	p_2	p_3	p_4	u_q	U_q
0	0.208461	0.481681	0.420538	0.859182	0.603430	0.435408	0.0	0.0	3.494901	3.467
1	0.781368	0.692829	0.969659	0.032401	0.000000	0.000000	0.0	1.0	3.266885	3.328
2	0.546571	0.964307	0.063893	0.383576	0.389439	0.491920	0.0	0.0	4.659477	4.544
3	0.930557	0.381203	0.665347	0.999155	0.145812	0.480583	0.0	0.0	3.470172	3.454
4	0.309831	0.129804	0.346928	0.770327	0.566760	0.039395	0.0	0.0	2.878247	2.886

WHICH IS THE BEST MEMORY ONE STRATEGY?

ARE THEIR LIMITATIONS TO MEMORY ONE STRATEGIES?


memory one strategy
complex strategy

	q_1	q_2	q_3	q_4	p_1	p_2	p_3	p_4	u_q	U_q
0	0.208461	0.481681	0.420538	0.859182	0.603430	0.435408	0.0	0.0	3.494901	3.467
1	0.781368	0.692829	0.969659	0.032401	0.000000	0.000000	0.0	1.0	3.266885	3.328
2	0.546571	0.964307	0.063893	0.383576	0.389439	0.491920	0.0	0.0	4.659477	4.544
3	0.930557	0.381203	0.665347	0.999155	0.145812	0.480583	0.0	0.0	3.470172	3.454
4	0.309831	0.129804	0.346928	0.770327	0.566760	0.039395	0.0	0.0	2.878247	2.886

This repository Search Pull requests Issues Marketplace Explore

sympy / **sympy**

Unwatch 316 | Star 4,837 | Fork 2,285

Code | Issues 2,763 | Pull requests 560 | Projects 0 | Wiki | Insights

implementation of multivariate resultants. #14370

Edit

Merged **jksuom** merged 19 commits into `sympy:master` from `Nikoleta-v3:multivariate-resultants` on 30 Mar

Conversation 63 | Commits 19 | Checks 0 | Files changed 6 | +3,033 −0

Nikoleta-v3 commented on 2 Mar — Contributor

Brief description

Adds a new file `multivariate_resultants` that contains two classes.
These classes are implementations of the following multivariate resultants:

- Dixons
- Macaulay

They are a natural follow up from the resultants implemented withing the library:

- subresultants_qq_zz.py

Resultants are used to indentify if polynomials have common roots. The multivariate version is for multivariate systems.

Other comments

Tests have been implemented and are currently passing. I added relevant literature and some notebooks with examples in the example/notebooks/ directory.

References to other Issues or PRs

Reviewers
- asmeurer
- smichr
- jksuom
- normalhuman

Assignees
No one assigned

Labels
PR: sympy's turn

Projects
None yet

Milestone
No milestone

Limitations of memory size on the Iterated Prisoner's dilemma. (In preperation)

@NikoletaGlyn
https://github.com/Nikoleta-v3