

Statistical Contribution To Drug Discovery

A Fast Exchange Algorithm For Designing Focused Libraries In Lead Optimisation

Céline Le Bailly de Tillegem
Institute of Statistics (U.C.L.)

Joined work with B. Govaerts (U.C.L. - Belgium),
B. Beck and B. Boulanger (Eli Lilly - Belgium)



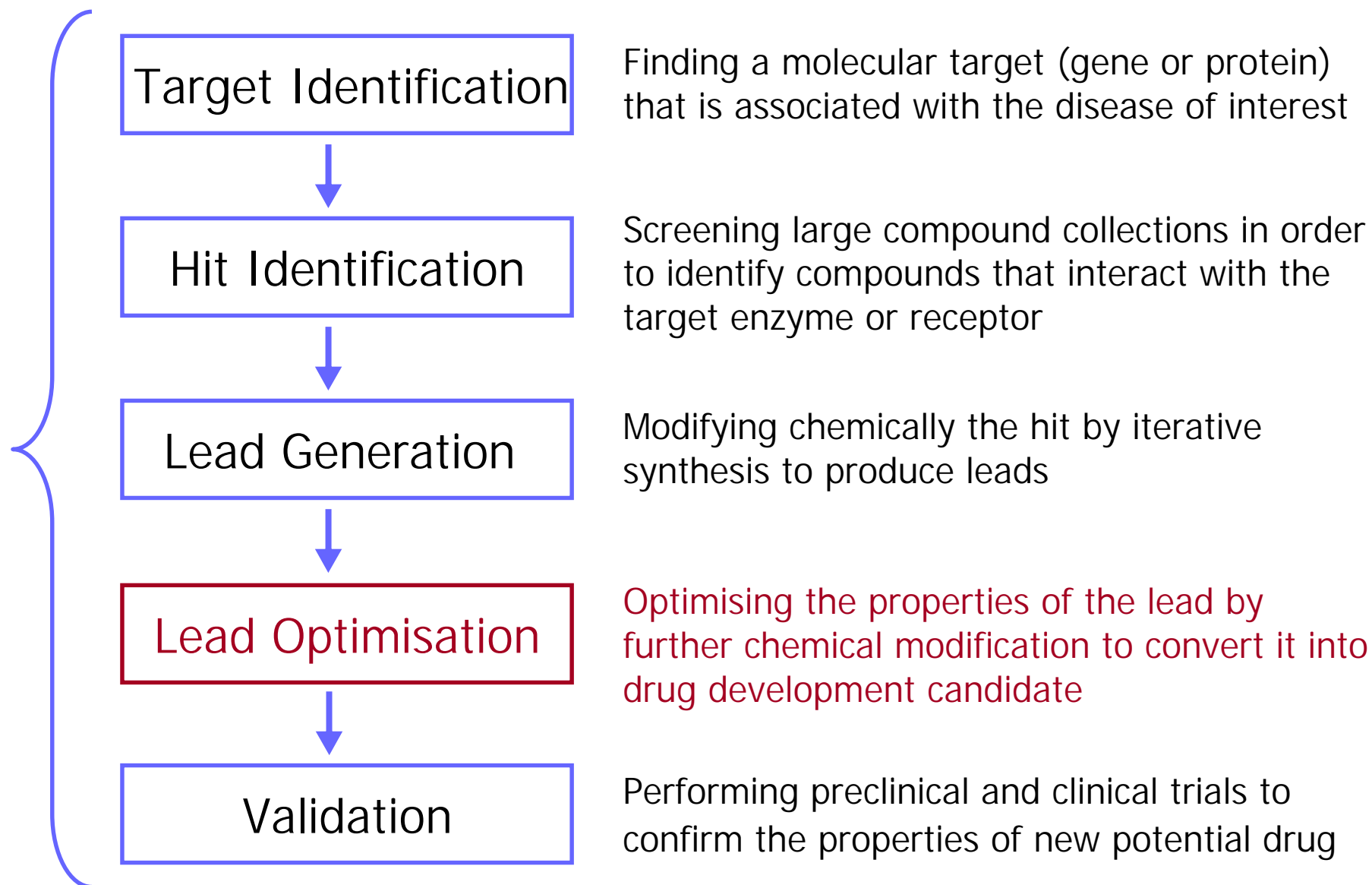
Outline Of The Talk

- Context of the research
 - Drug discovery process
 - Lead optimisation
 - Focused library design
- Our contribution: Weighting Exchange Algorithm for Library Design
 - Principles of W.E.A.L.D.
 - Application on a real library
- Comparison with 2 other existing methods
 - Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)
 - Piccolo Algorithm (Zheng & all, 2000)
- Future research

Outline Of The Talk

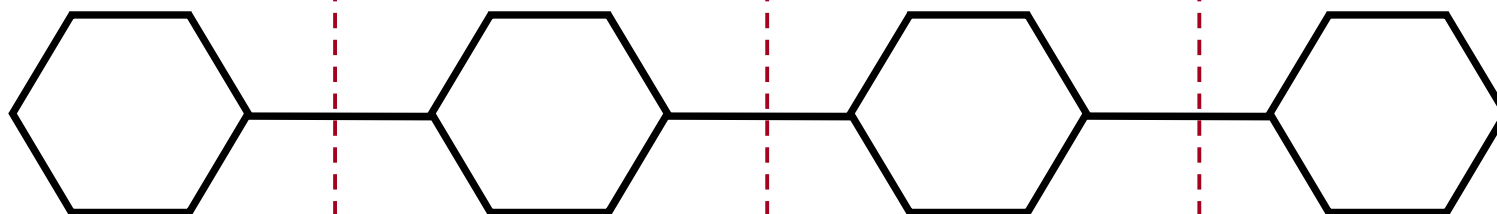
- Context of the research
 - Drug discovery process
 - Lead optimisation
 - Focused library design

Drug Discovery Process



Lead Optimisation (1)

LEAD



r-group A
12 reagents

a1 a2 a3 a4
a5 a6 a7 a8
a9 a10 a11 a12



r-group B
19 reagents

b1 b2 b3 b4 b5 b6
b7 b8 b9 b10 b11
b12 b13 b14 b15
b16 b17 b18 b19



r-group C
12 reagents

c1 c2 c3 c4
c5 c6 c7 c8
c9 c10 c11 c12

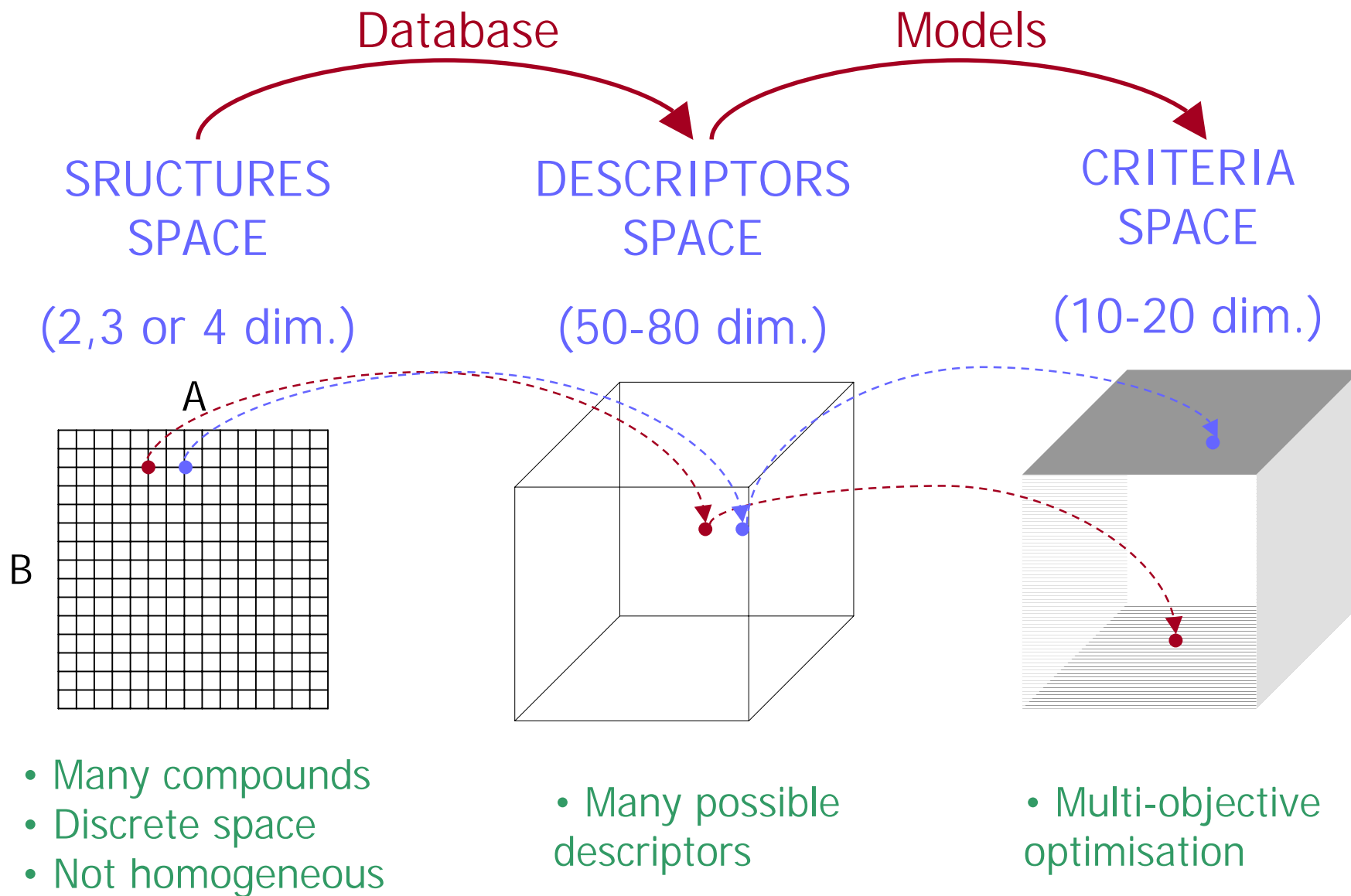


r-group D
41 reagents

d1 d2 d3 d4 d5 d6 d7
d8 d9 d10 d11 d12 d13
d14 d15 d16 d17 d18
d19 d20 d21 d22 d23
d24 d25 d26 d27 d28
d29 d30 d31 d32 d33
d34 d35 d36 d37 d38
d39 d40 d41

⇒ Combinatorial library with $12 \times 19 \times 12 \times 41 = 112176$ possible combinations/compounds

Lead Optimisation (2)



Focused Library Design - Objective

PROBLEM : huge time to compute the properties of interest for the 112176 possible compounds

OBJECTIVE : select a small subset of reagents in each r-group to obtain a sub-library of reasonable size

- exploratory - diverse library
- focused - targeted library ← Multi-objective optimisation

r-group A
12 reagents

a1 a2 a3 a4
a5 a6 a7 a8
a9 a10 a11 a12

r-group B
19 reagents

b1 b2 b3 b4 b5 b6
b7 b8 b9 b10 b11
b12 b13 b14 b15
b16 b17 b18 b19

r-group C
12 reagents

c1 c2 c3 c4
c5 c6 c7 c8
c9 c10 c11 c12

r-group D
41 reagents

d1 d2 d3 d4 d5 d6 d7
d8 d9 d10 d11 d12 d13
d14 d15 d16 d17 d18
d19 d20 d21 d22 d23
d24 d25 d26 d27 d28
d29 d30 d31 d32 d33
d34 d35 d36 d37 d38
d39 d40 d41

⇒ $3 \times 3 \times 3 \times 3 = 81$ combinations
BUT there are about 5×10^{11} such sub-libraries

Outline Of The Talk

- Context of the research
 - Steps in drug discovery
 - Lead optimisation
 - Focused library design
- Our contribution: Weighting Exchange Algorithm for Library Design
 - Principles of W.E.A.L.D.
 - Application on a real library

Principles of W.E.A.L.D.

- Select a sub-library *at random*
- *Iterate* using exchanges to improve the quality of the current sub-library
- *Stop* if there is no more possible exchange

➔ How to quantify the quality of a sub-library ?

➔ How to perform exchanges ?

How to quantify the quality of a sub-library ?

- The quality of a **combination** is quantified by a **desirability** value laying between 0 and 1 that summarises the 8 properties

N	R_A	R_B	R_D	R_C	P1	P2	P3	P4	P5	P6	P7	P8	DESIRABILITY
1	a7	b16	d3	c3	-1.76	-2.48	-3.90	-3.57	-0.55	-3.50	0.97	-15.09	0.6908
2	a1	b3	d8	c2	-1.80	-2.65	-2.39	-1.45	-0.53	-3.05	0.93	-31.89	0.6023
3	a11	b1	d3	c8	-1.29	-1.59	-1.89	-0.69	-0.92	-2.06	1.00	11.63	0.5956
4	a8	b15	d18	c8	-1.20	-2.54	-2.88	-1.14	-1.13	-2.27	0.77	-10.90	0.5910
5	a3	b2	d13	c1	-2.48	-2.34	-2.73	-1.31	-0.29	-2.52	0.65	-33.49	0.5406
6	a2	b14	d3	c4	-2.13	-2.46	-2.56	-1.76	-1.47	-2.78	0.74	-39.79	0.5129
7	a8	b6	d30	c6	-2.08	-2.86	-2.94	-1.31	-1.22	-2.29	0.76	-66.83	0.4891
8	a10	b1	d40	c7	-1.60	-2.25	-2.17	-0.84	-1.62	-2.66	0.86	-72.06	0.4732
9	a6	b19	d17	c5	-2.13	-2.56	-3.37	-0.92	-2.15	-1.64	0.60	-22.29	0.4498
10	a6	b17	d37	c7	-1.77	-2.26	-3.21	-1.59	-2.24	-2.72	0.57	-66.78	0.4310
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

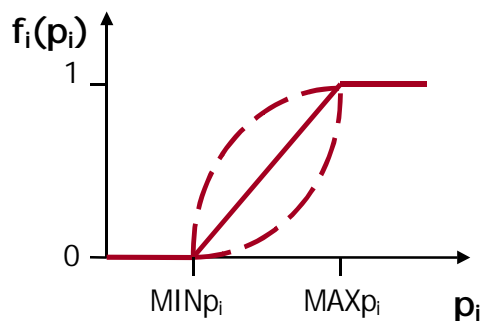
How to quantify the quality of a sub-library ?

$$\text{Desirability} = \prod_{i=1}^8 [f_i(p_i)]^{w_i}$$

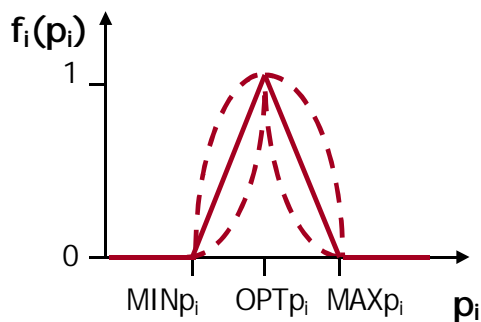
weight

i^{th} property

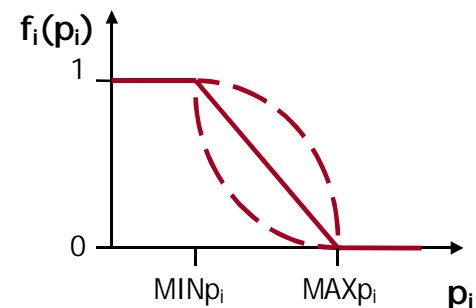
function that maps the i^{th} property between 0 and 1



maximise P_i

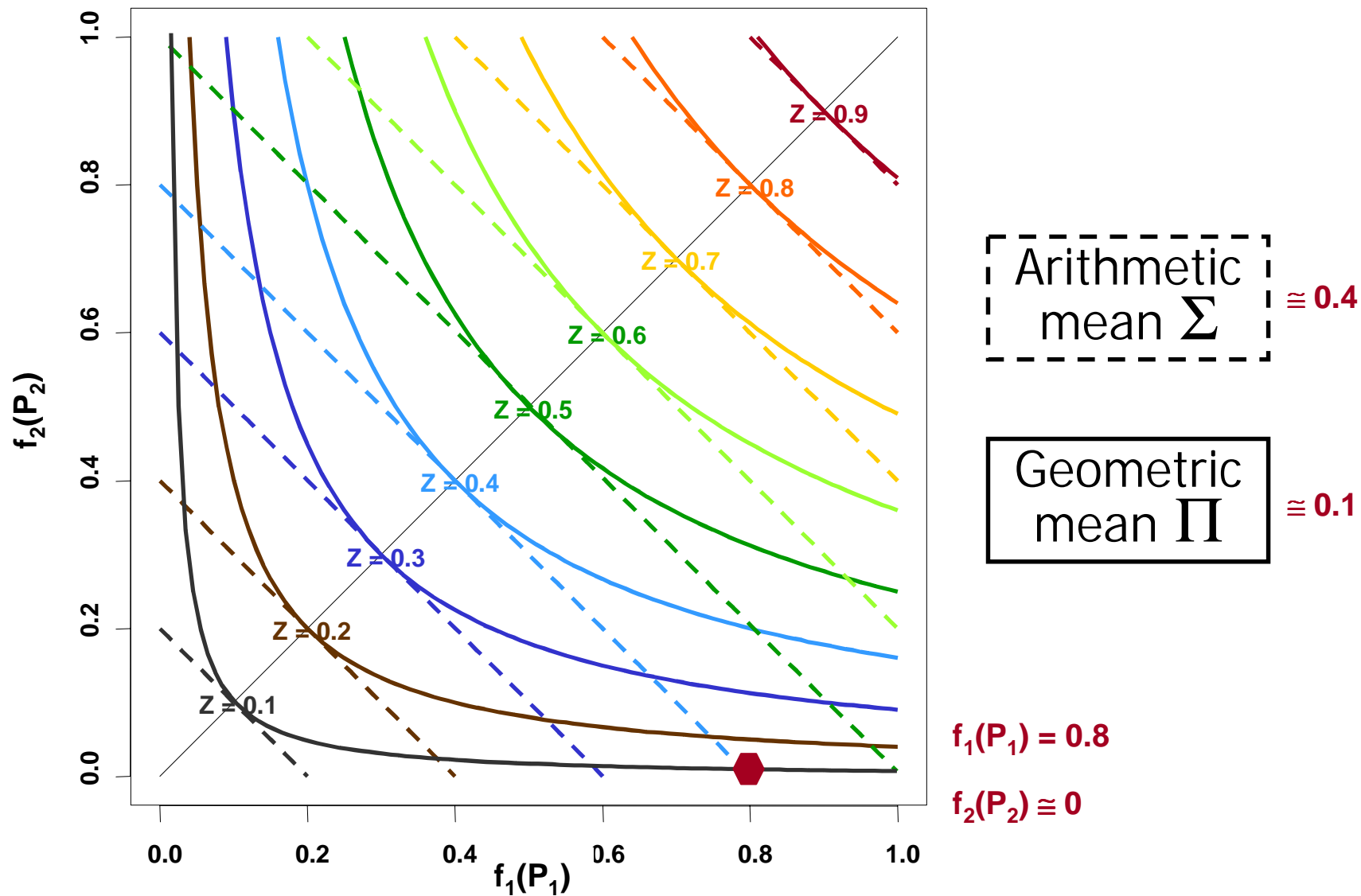


reach an
optimal value



minimise P_i

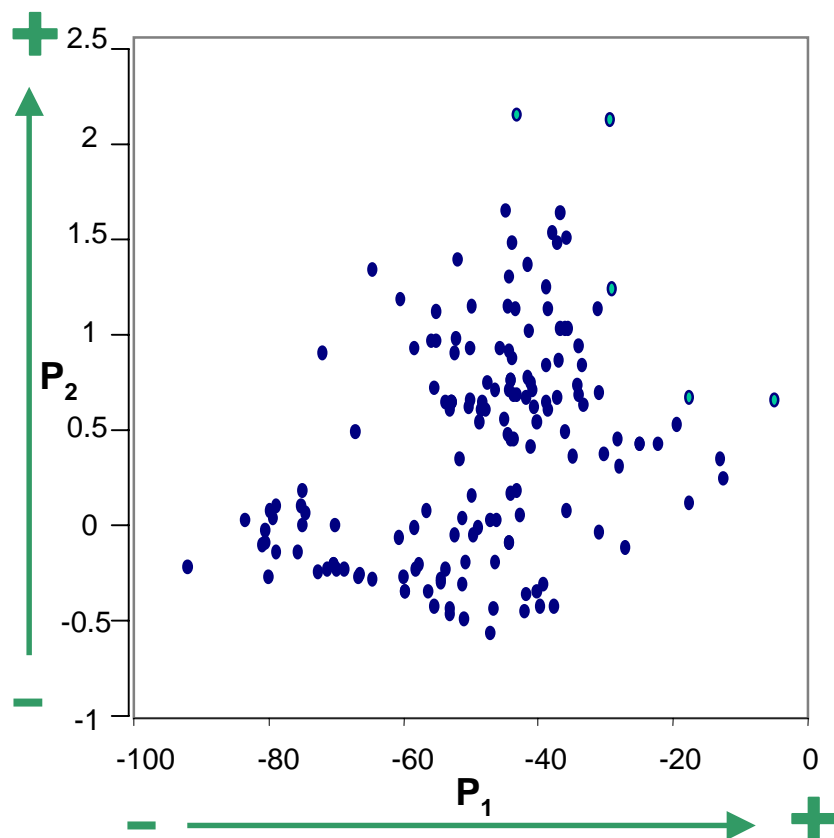
How to quantify the quality of a sub-library ?



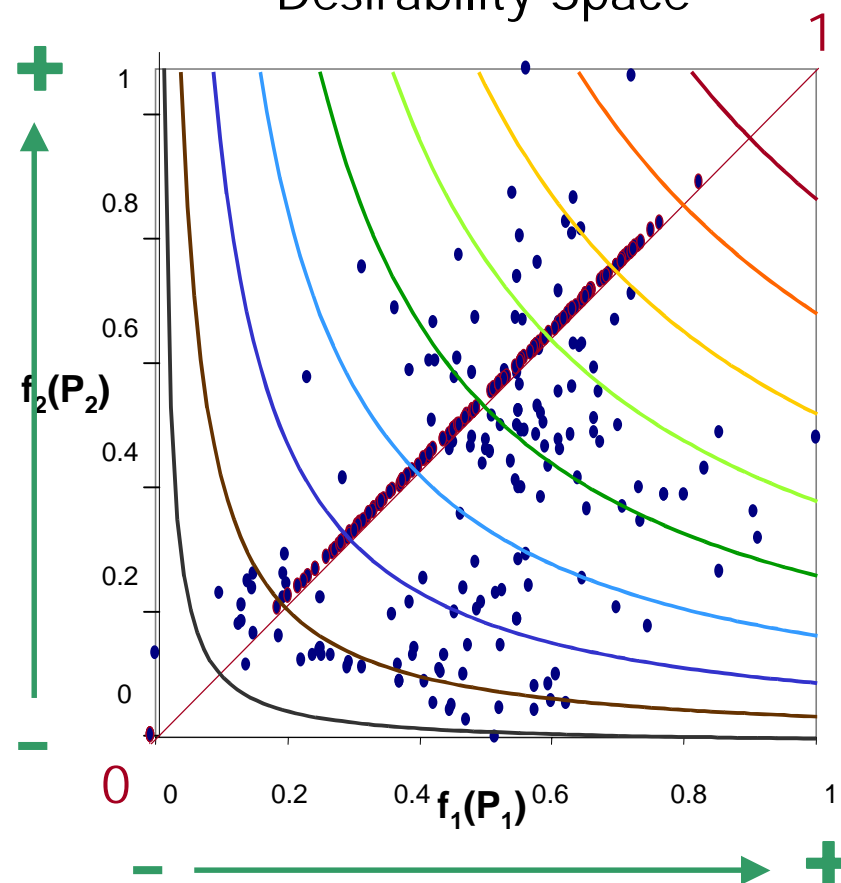
How to quantify the quality of a sub-library ?

Example in 2 dimensions

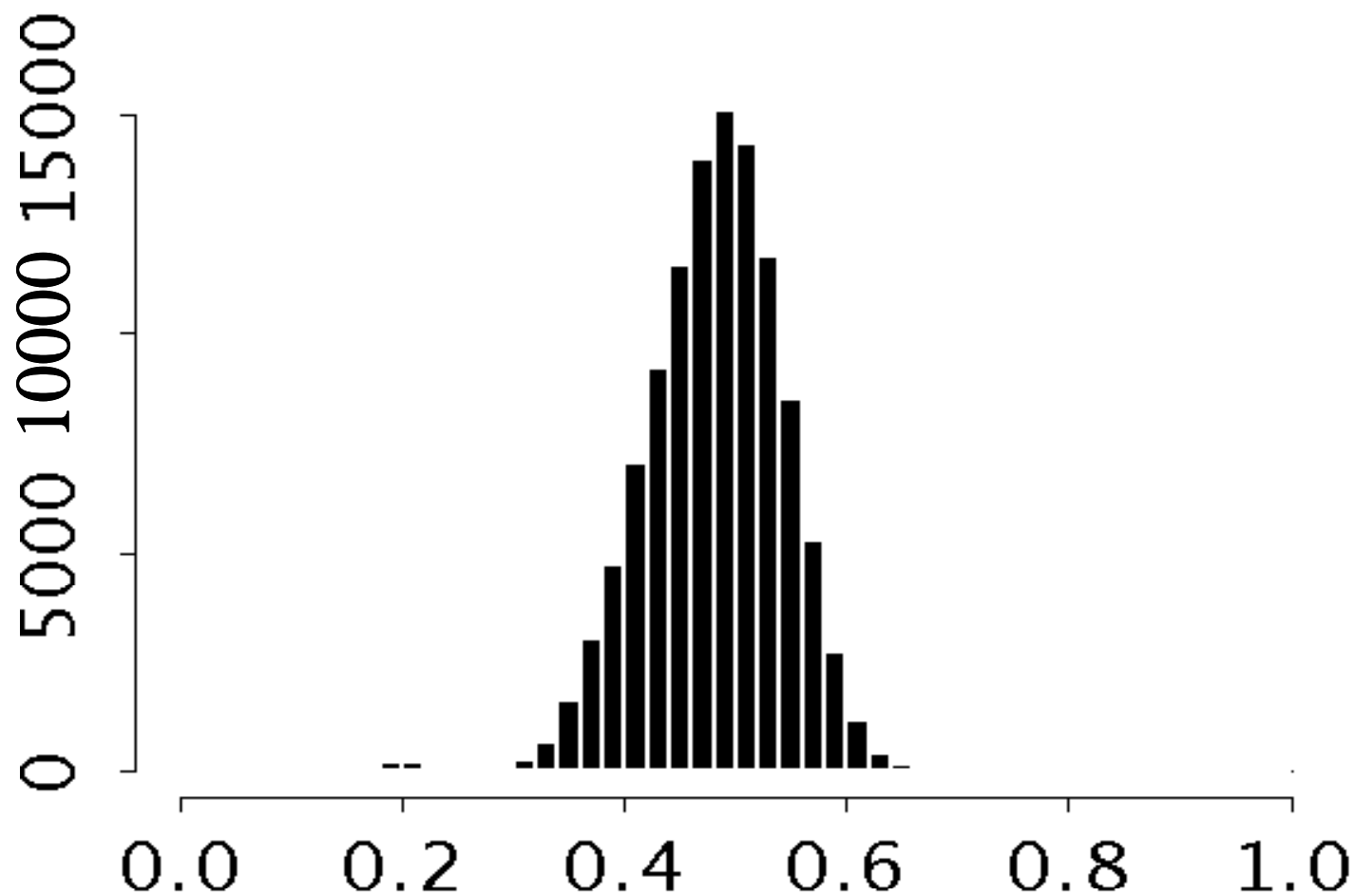
Criteria Space



Desirability Space



How to quantify the quality of a sub-library ?



Desirabilities of the 112176 compounds of the whole combinatorial library

How to quantify the quality of a sub-library ?

- The quality of a **combination** is quantified by a **desirability** value laying between 0 and 1 that summarises the 8 properties

$$\text{Desirability} = \prod_{i=1}^8 [f_i(p_i)]^{w_i}$$

- The overall quality of a **sub-library** is quantified by a **loss** function that summarises the differences between the 81 desirabilities and their ideal value 1

$$\text{Loss} = \frac{1}{81} \sum_{k=1}^{81} (1 - \text{Desirability}_k)^2$$

Principles of W.E.A.L.D.

- Select a sub-library *at random*
- *Iterate* using exchanges to improve the quality of the current sub-library
- *Stop* if there are no more possible exchanges

➔ How to quantify the quality of a sub-library ?

➔ How to perform exchanges ?

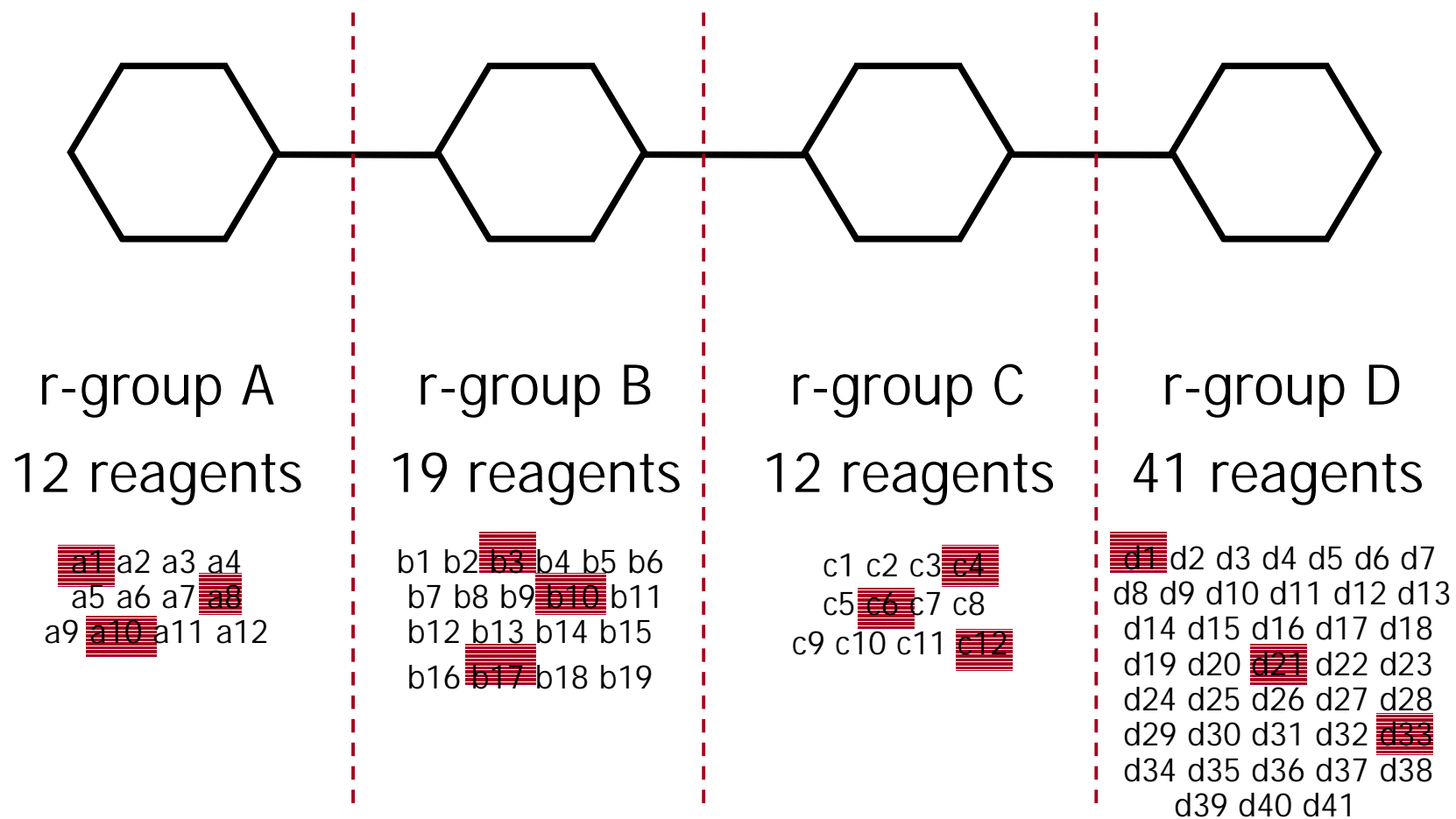
How to perform exchanges ?

- Select a reagent to be **ejected** from the current sub-library
 - ➔ select the “worst” one
- Select a **new** reagent to replace the ejected one
 - ➔ select according to probabilities for reagents to enter the sub-library
- Apply the **exchange** if the loss decreases



When **no more possible exchange** with reagents that have a probability of selection > 0

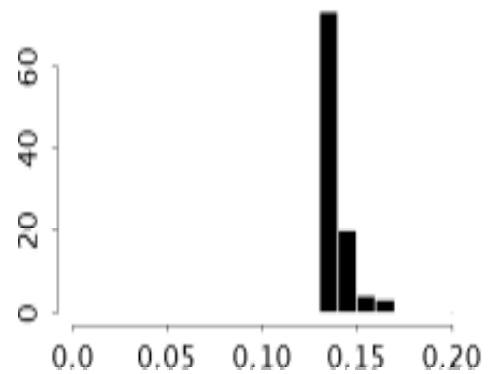
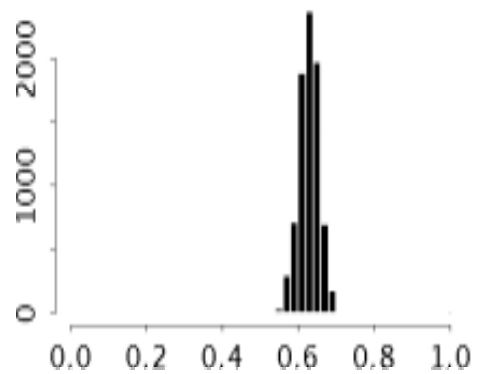
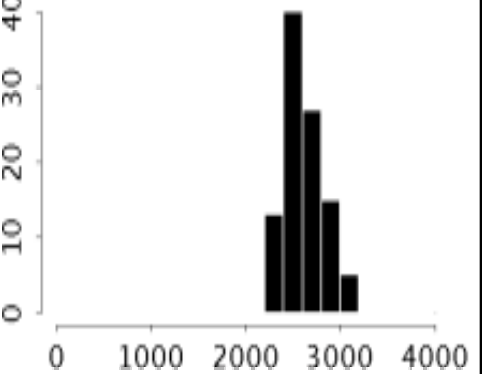
Application of W.E.A.L.D. on a real library



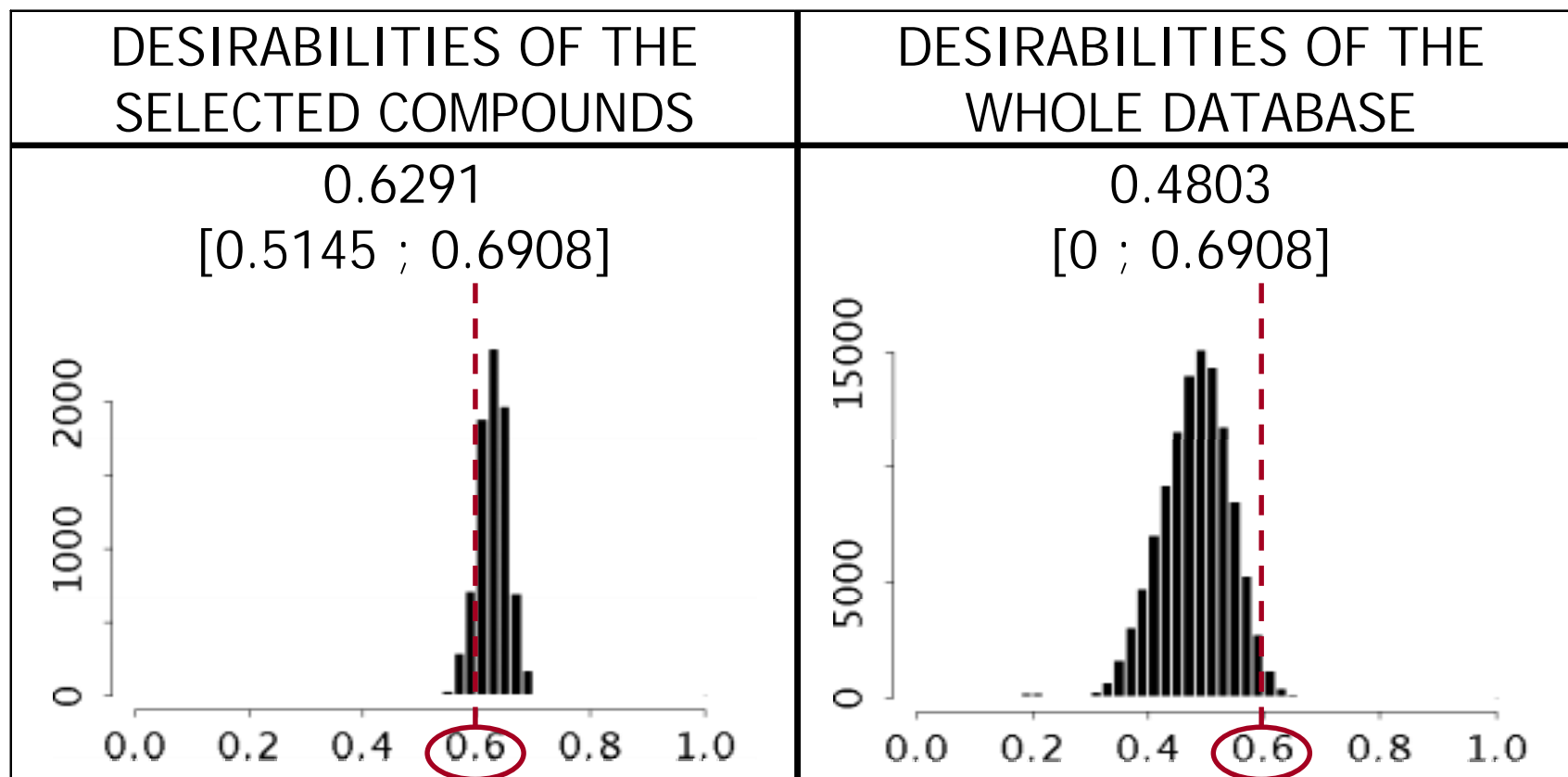
⇒ Which sub-library $3 \times 3 \times 3 \times 3$ to choose ?

Application of W.E.A.L.D. on a real library

Application of W.E.A.L.D. algorithm on a real database
 (4 r-groups - 112176 compounds)
 with 100 different initial sub-libraries 3×3×3×3

	LOSS OF THE RESULTING SUB-LIBRARIES	DESIRABILITIES OF THE SELECTED COMPOUNDS	NUMBER OF DESIRABILITIES CALCULATIONS
Mean [min;max] std. dev.	0.1382 [0.1302 ; 0.1619] 0.0075	0.6291 [0.5145 ; 0.6908] 0.0255	2613 (2.33%) [2220 ; 3114] 199.99
			

Application of W.E.A.L.D. on a real library



Outline Of The Talk

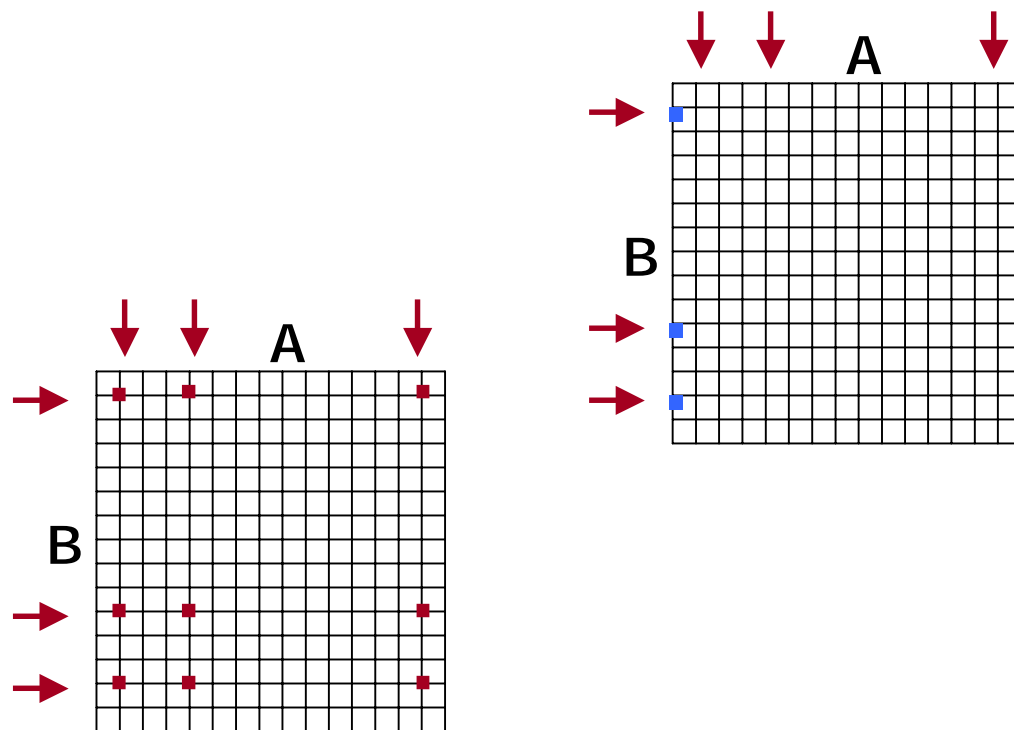
- Context of the research
 - Steps in drug discovery
 - Lead optimisation
 - Focused library design
- Our contribution: Weighting Algorithm for Library Design
 - Principles of W.E.A.L.D.
 - Application on a real library
- Comparison with 2 other existing methods
 - Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)
 - Piccolo Algorithm (Zheng et Al., 2000)

Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)

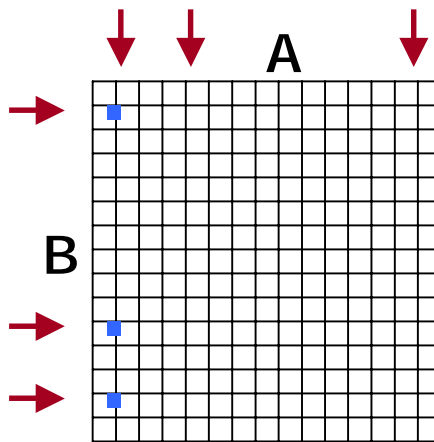
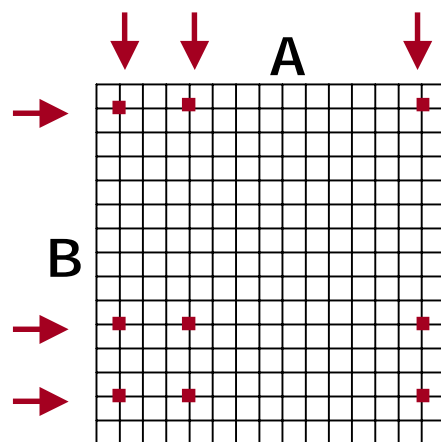
- **PRINCIPLES**

- ① Initialise a sub-library $3 \times 3 \times 3 \times 3$ at random, S , and evaluate its quality
- ② Randomise the order of the reagents in each r-group
- ③ For each r-group ($i=1,2,3,4$)
 - For each reagent of the r-group ($j=1,2,\dots,N_i$)
 - Evaluate the quality of the sub-library obtained by combining this reagent with the selected reagents of the current library

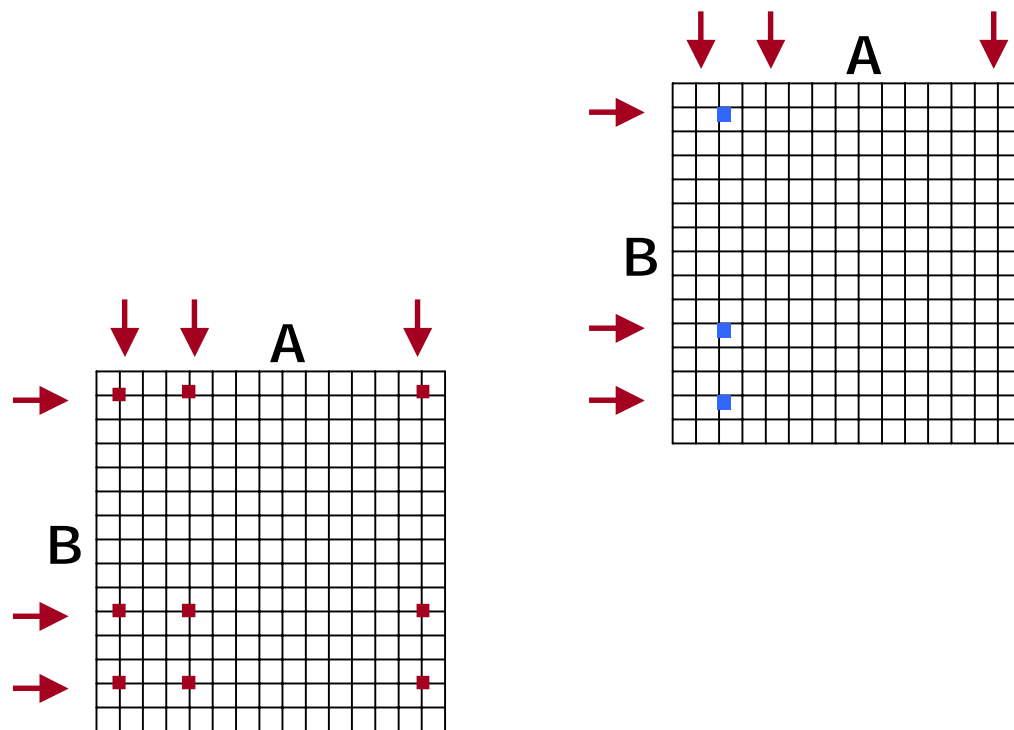
Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)



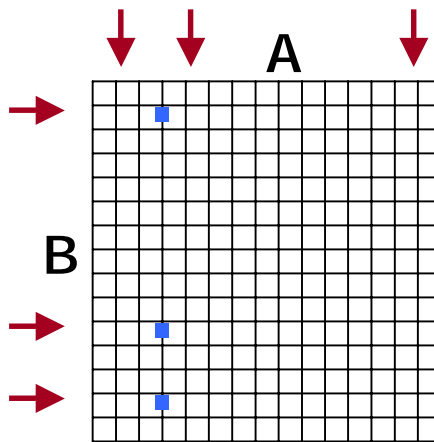
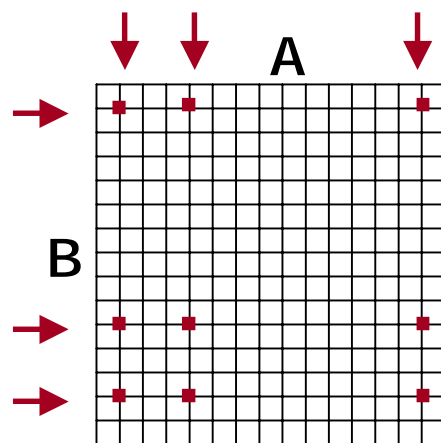
Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)



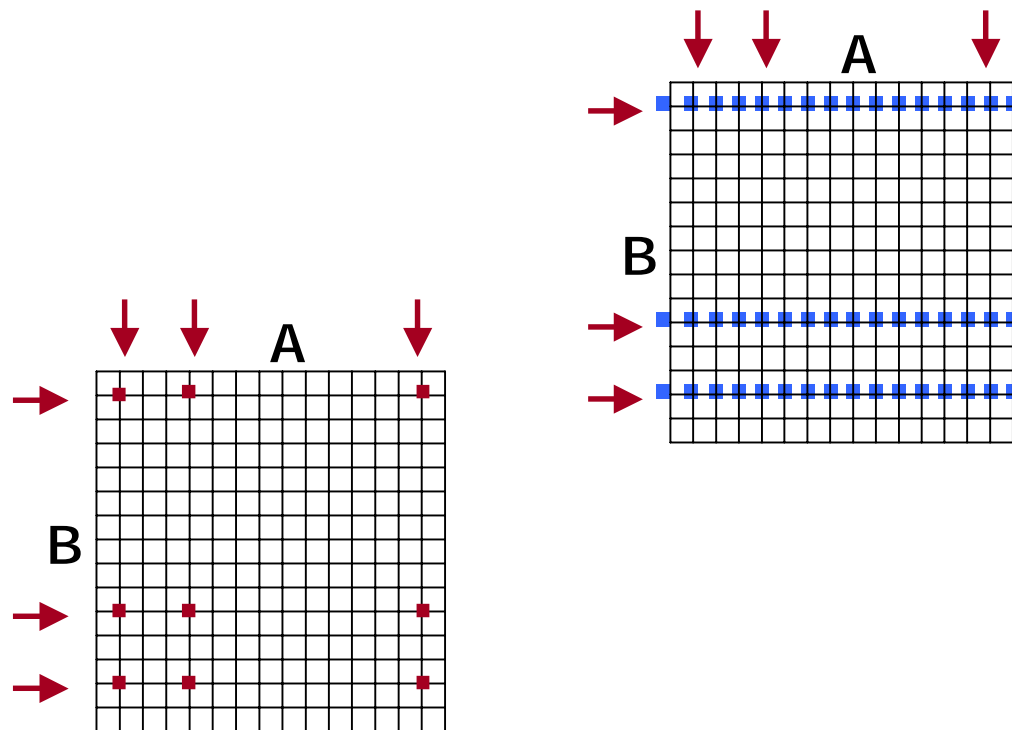
Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)



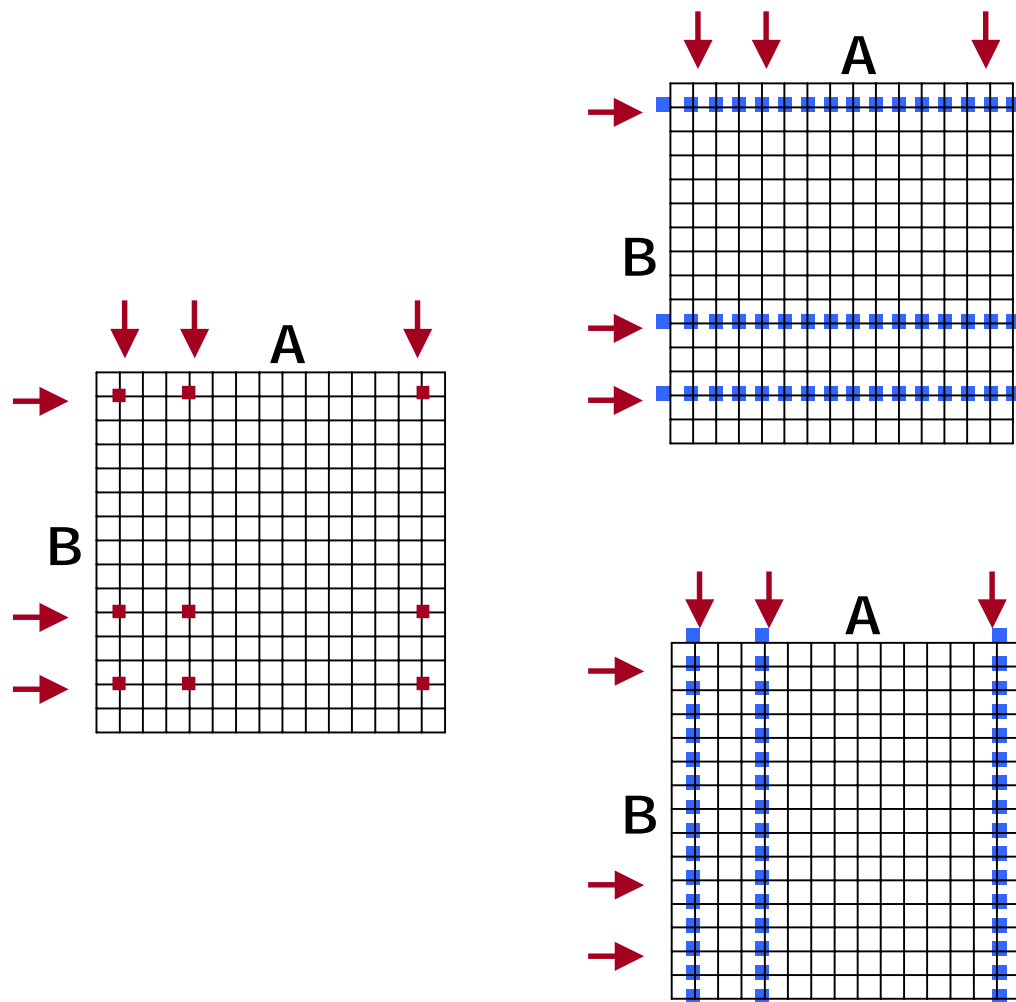
Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)



Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)



Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)

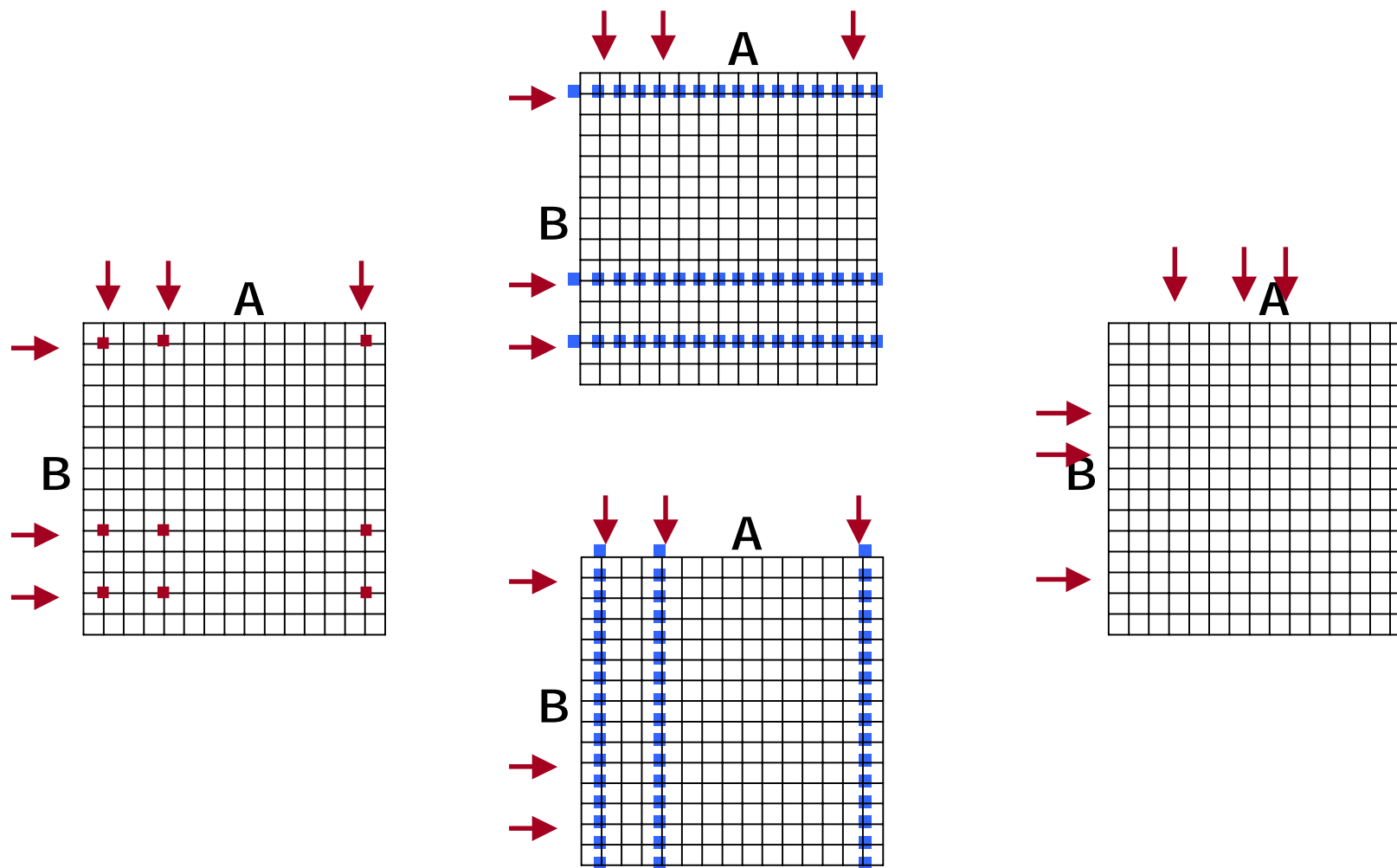


Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)

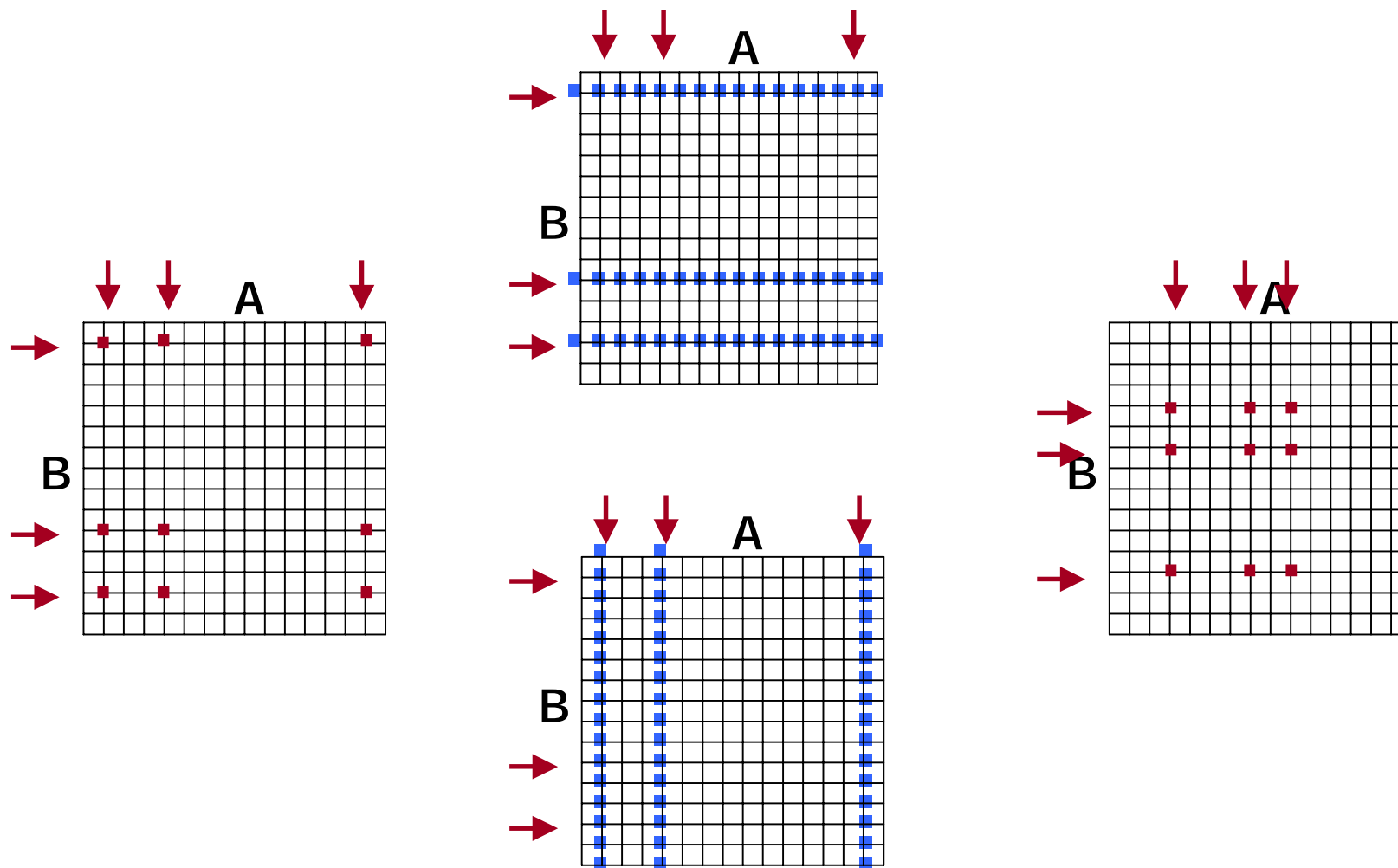
- **PRINCIPLES**

- ① Initialise a sub-library $3 \times 3 \times 3 \times 3$ at random, S , and evaluate its quality
- ② Randomise the order of the reagents in each r-group
- ③ For each r-group ($i=1,2,3,4$)
 - For each reagent of the r-group ($j=1,2,\dots,N_i$)
 - Evaluate the quality of the sub-library obtained by combining this reagent with the selected reagents of the current library
- ④ In each r-group sort the reagents in descending order of quality and select the 3 best ones. Combine those selected reagents and evaluate the quality of the resulting sub-library

Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)



Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)

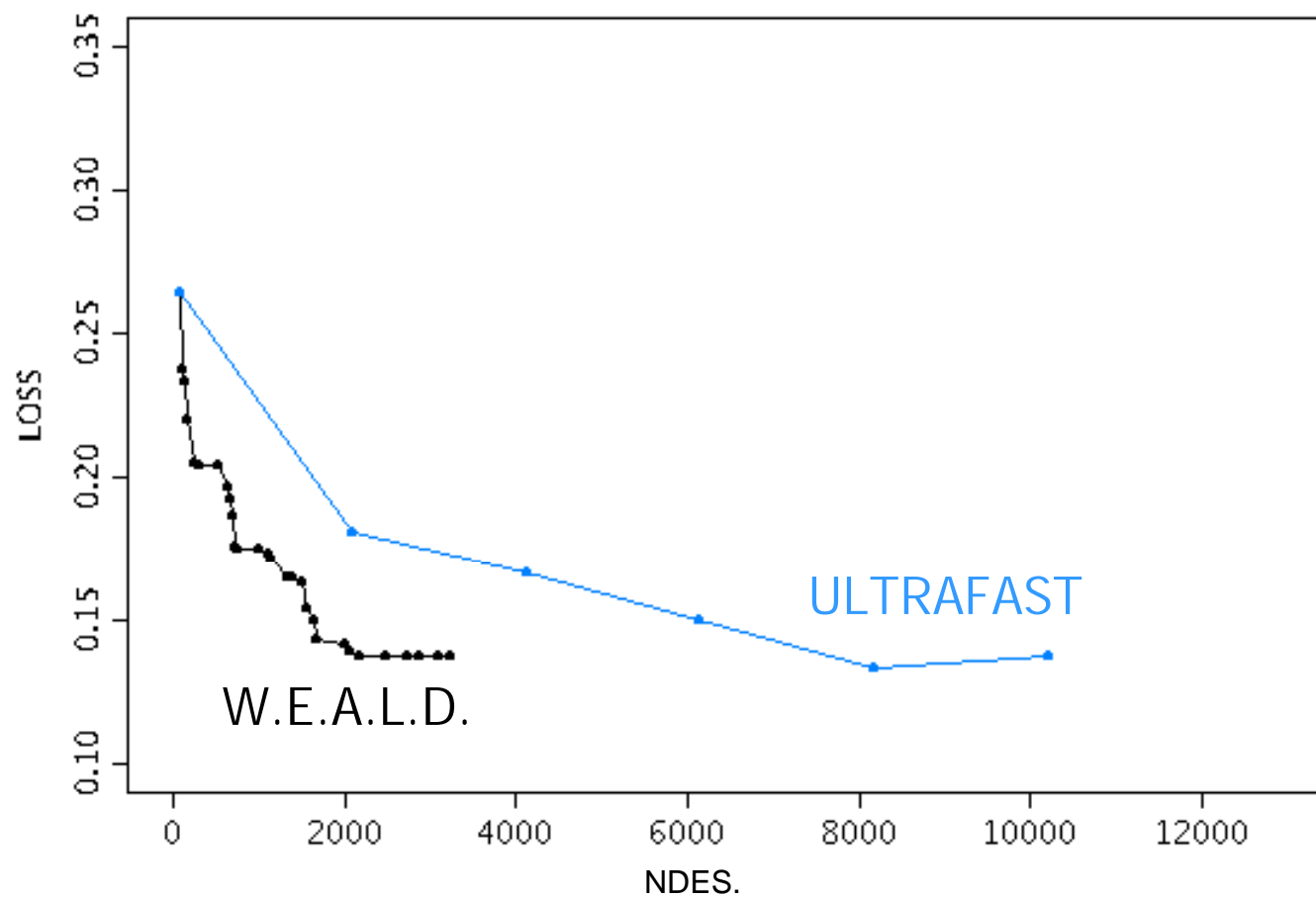


Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)

- **APPLICATION** of the 2 methods on a real library with 50 different initial sub-libraries 3×3×3×3

	LOSS OF THE RESULTING SUB-LIBRARIES	DESIRABILITIES OF THE SELECTED COMPOUNDS	NUMBER OF DESIRABILITIES CALCULATIONS
Mean [min;max] std. dev. ULTRAFAST ALGORITHM	0.1443 [0.1302 ; 0.1975] 0.0156 	0.6217 [0.4802 ; 0.6908] 0.0347 	7857 (7%) [4131 ; 12231] 2599
Mean [min;max] std. Dev. W.E.A.L.D ALGORITHM	0.1375 [0.1302 ; 0.1616] 0.0077 	0.6301 [0.5145 ; 0.6908] 0.0258 	2633 (2.35%) [2240 ; 3142] 230

Comparison with Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)



Comparison with Piccolo Algorithm (Zheng, Hung, Saunders and Seibel, 2000)

- **PRINCIPLES**

- **Initialise** a sub-library at random
- **Iterate** using exchanges to improve the quality of the current sub-library
- **Stop** if there are no more possible exchanges or if more than r successive exchanges did not increase the quality of the current sub-library

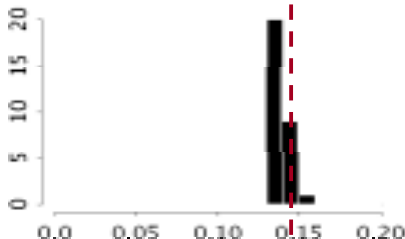
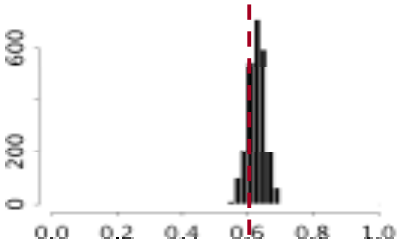
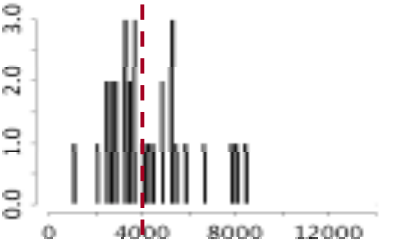
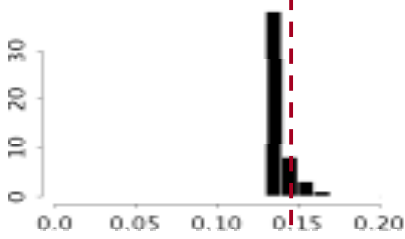
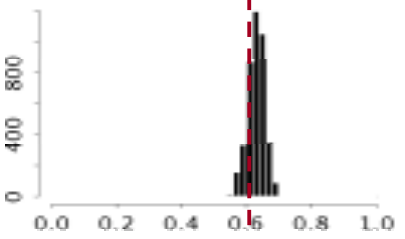
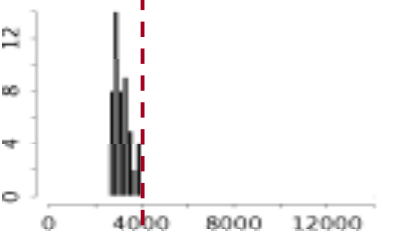
Comparison with Piccolo Algorithm (Zheng, Hung, Saunders and Seibel, 2000)

- **PRINCIPLES**

- Initialise a sub-library at random
- Iterate using exchanges to improve the quality of the current sub-library
- Stop if there are no more possible exchanges or if more than r successive exchanges did not increase the quality of the current sub-library

Comparison with Piccolo Algorithm (Zheng, Hung, Saunders and Seibel, 2000)

- **APPLICATION** of the 2 methods on a real library with 30 different initial sub-libraries 3×3×3×3

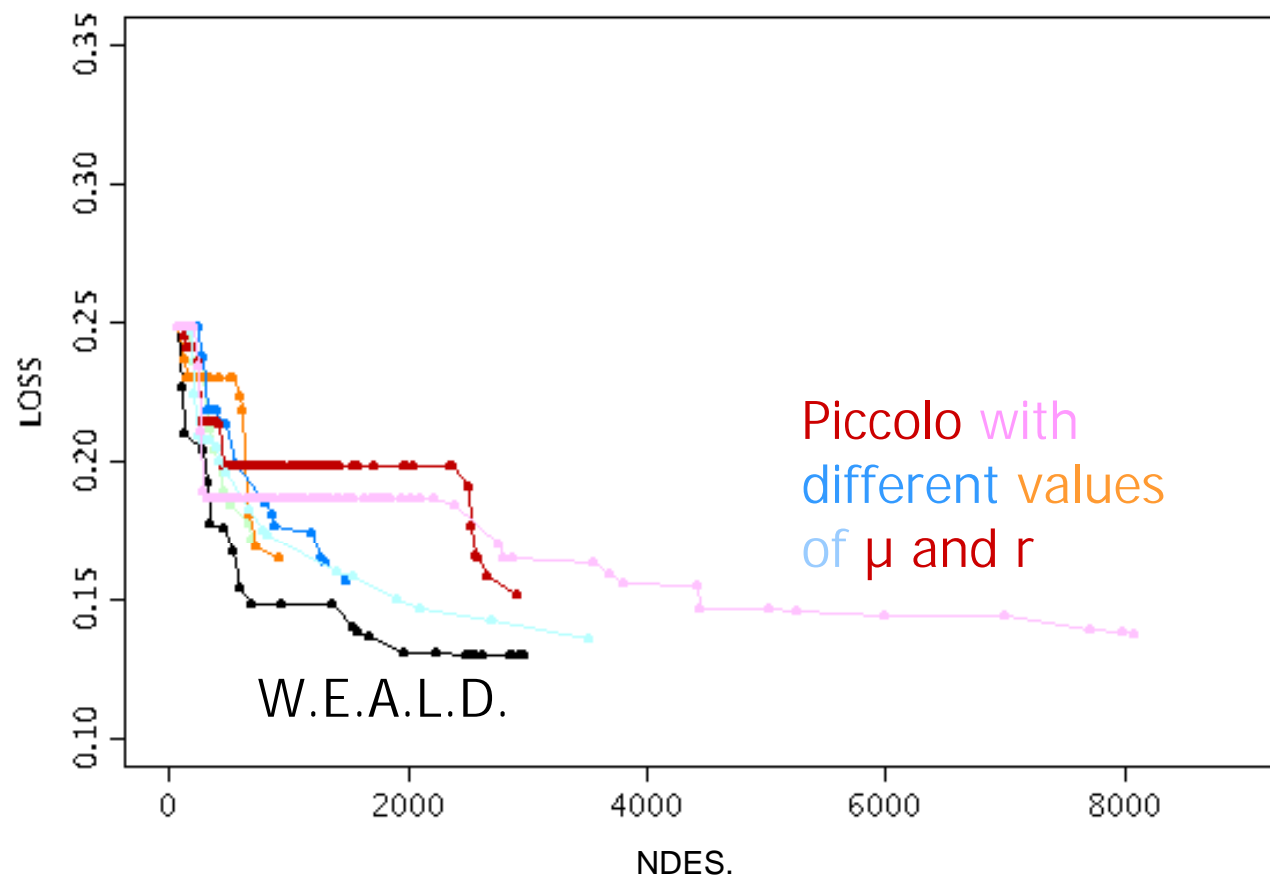
	LOSS OF THE RESULTING SUB-LIBRARIES	DESIRABILITIES OF THE SELECTED COMPOUNDS	NUMBER OF DESIRABILITIES CALCULATIONS
Mean [min;max] std. dev. PICCOLO ALGORITHM $\mu = 0.9$ $r = 41$	0.1383 [0.1302;0.1532] 0.0063 	0.6290 [0.5296;0.6908] 0.0265 	4298 (3.83%) [1161;8532] 1790 
Mean [min;max] std. Dev. W.E.A.L.D. ALGORITHM	0.1375 [0.1302;0.1616] 0.0077 	0.6301 [0.5145;0.6908] 0.0258 	2633 (2.35%) [2240;3142] 230 

Comparison with Piccolo Algorithm (Zheng, Hung, Saunders and Seibel, 2000)

- **APPLICATION** of the 2 methods on a real library with 30 different initial sub-libraries 3×3×3×3

	LOSS OF THE RESULTING SUB-LIBRARIES	DESIRABILITIES OF THE SELECTED COMPOUNDS	NUMBER OF DESIRABILITIES CALCULATIONS
<p>Mean [min;max] std. dev.</p> <p>PICCOLO ALGORITHM $\mu = 0.6$ $r = 12$</p>	<p>0.1723 [0.1413;0.2117] 0.0174</p>	<p>0.5868 [0.4319;0.6908] 0.0390</p>	<p>921 (0.82%) [513;1917] 361</p>
<p>Mean [min;max] std. Dev.</p> <p>W.E.A.L.D. ALGORITHM</p>	<p>0.1375 [0.1302;0.1616] 0.0077</p>	<p>0.6301 [0.5145;0.6908] 0.0258</p>	<p>2633 (2.35%) [2240;3142] 230</p>

Comparison with Piccolo Algorithm (Zheng, Hung, Saunders and Seibel, 2000)



Conclusion

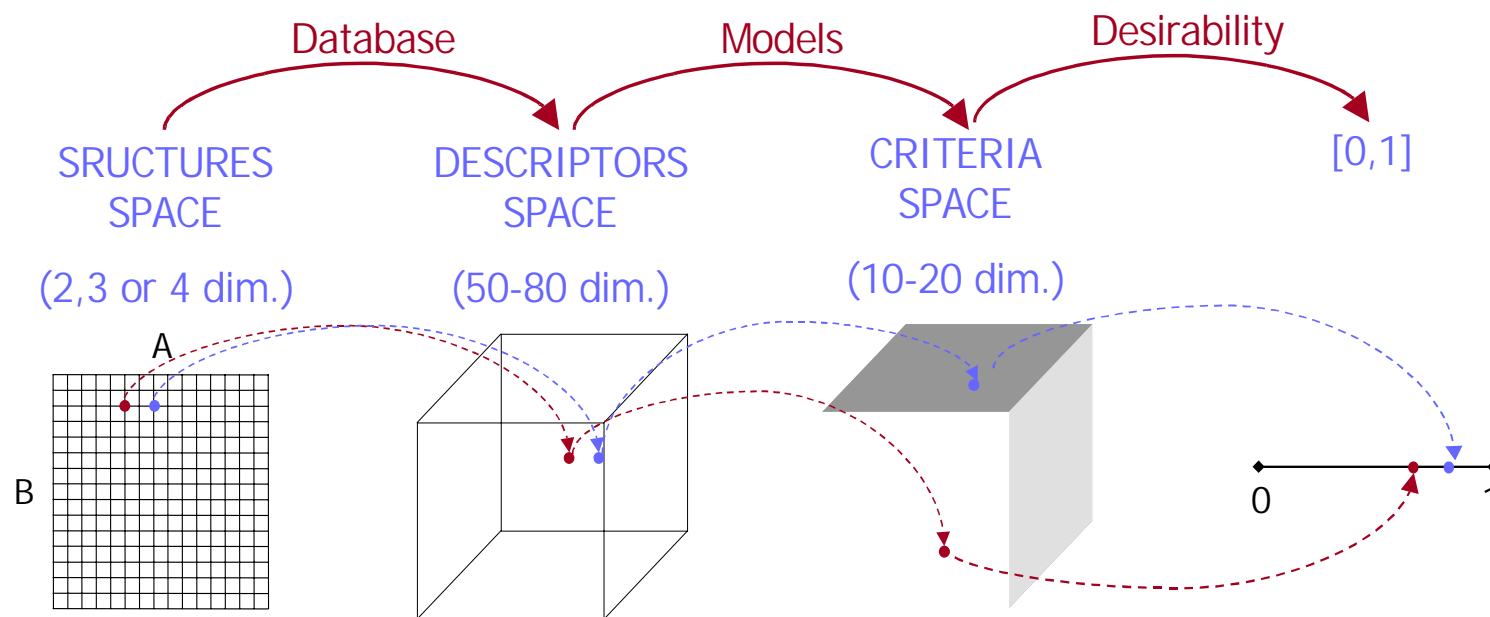
Our algorithm

- is simple
- is fast
- gives “good” sub-libraries
- works well, whatever is the dimension of the whole library

Outline Of The Talk

- Context of the research
 - Steps in drug discovery
 - Lead optimisation
 - Focused library design
- Our contribution: Weighting Algorithm for Library Design
 - Principles of W.A.L.D.
 - Application on a real library
- Comparison with 2 other existing methods
 - Ultrafast Algorithm (Agrafiotis & Lobanov , 2000)
 - Piccolo Algorithm (Zheng et AL., 2000)
- Future research

Future research



- Design of Experiments : how to gather data (more diverse as possible) to estimate in-silico models
- Impact of prediction error on the desirabilities and on the optimisation process when combining different prediction models: how to take the prediction error into account

Some references...

Derringer G.C., Suich, R. (1980)

Simultaneous optimization of several response variables

Journal of Quality Technology, Vol 12, No 4, 214-219

Federov, V.V. (1972).

Theory of optimal experiments

Academic Press, NY.

W. Zheng, S. T. Hung, J. T. Saunders, and G. L. Seibel (2000)

PICCOLO: a tool for combinatorial library design via multicriterion optimisation

Pacific Symposium on Biocomputing 5:585-596

Dimitris K. Agrafiotis and Victor S. Lobanov (2000)

Ultrafast algorithm for designing focused combinatorial arrays

J. Chem. Inf. Comput. Sci. 2001, 40, 1030-1038