A new smart-pooling strategy for high-throughput screening: the Shifted Transversal Design

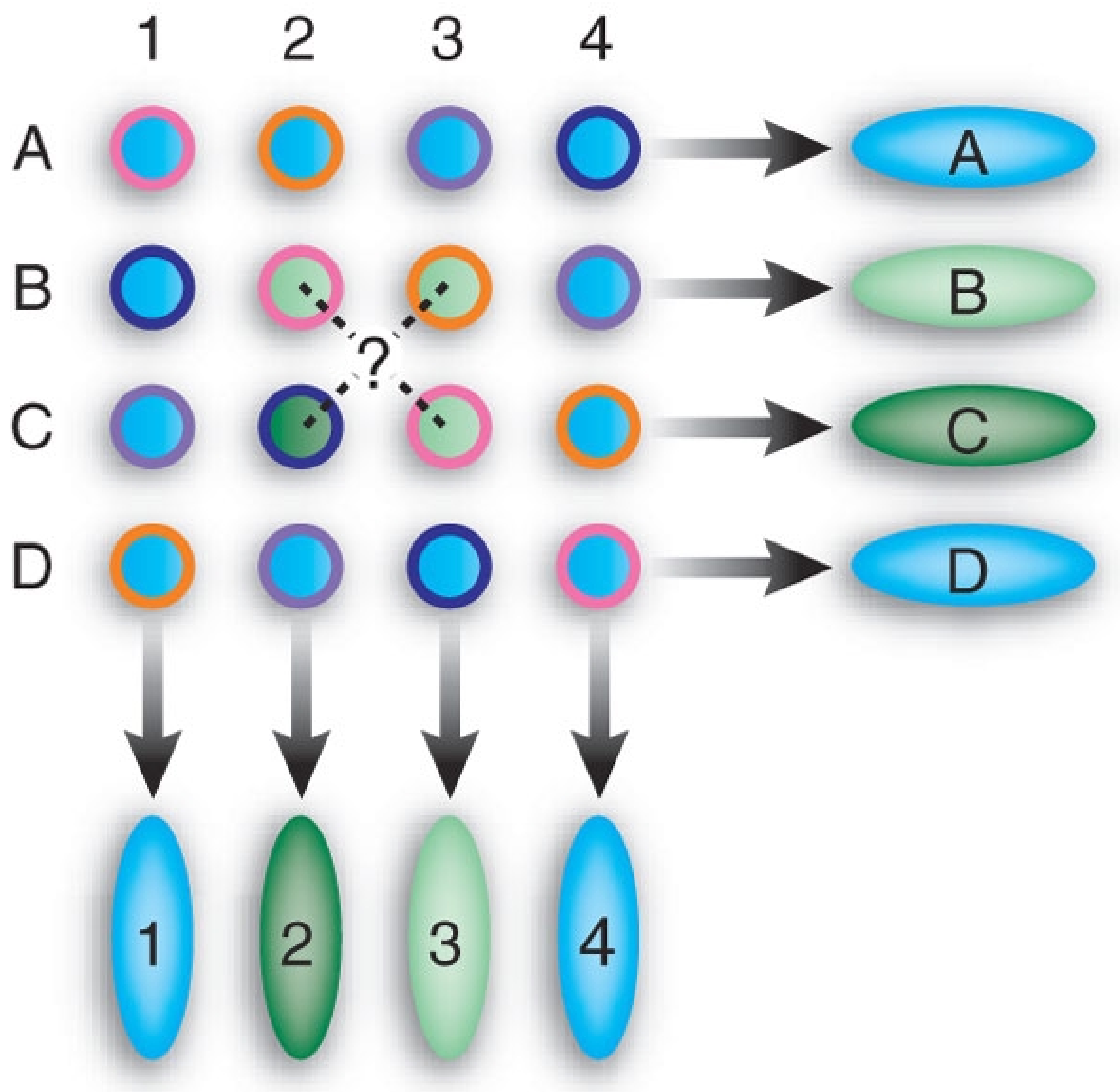
Nicolas Thierry-Mieg
CNRS / LSR-IMAG laboratory
Grenoble, France

DIMACS CGT Workshop, 17/05/2006

Context: systems biology

- Many high-throughput projects
  - basic **yes-or-no test** to a large collection of “objects”
  - **low-frequency positives**
  - **experimental noise**
- A natural solution: smart-pooling, provided that
  - objects are individually available
  - basic assay on pool of objects (OR: XOR is not available)
- Advantages:
  - Number of pools is small
  - Pools are redundant → error-correction
- Main difficulty: designing the pools
  - Non-adaptive designs
  - Specific constraints (e.g. pool size)

Example of smart-pooling: row and columns



(from: Thierry-Mieg N. Pooling in systems biology becomes smart. Nat Methods. 2006 Mar;3(3):161-2.)

Layout of the talk

- Biological context
- **Definition of STD**
- Properties
- Behavior and efficiency
- Application: protein-protein interaction mapping

STD: preliminary definitions

- Pooling problem (n,t,E):
  - $\mathcal{A}_n = \{A_0, \ldots, A_{n-1}\}$ set of Boolean variables ($n \approx 10^3$-10^6)
  - t = number of positives (≈1-10)
  - E = number of errors (≈1-40% of tests)

- Pool: subset of $\mathcal{A}_n$, value=OR

- Goal: build a set of **v** pools

  → v small

  → guarantee correction of errors & identification of positives

Matrix representation

v×n Boolean matrix: M(i,j) true $\Leftrightarrow$ pool i contains variable j

Example: n=9, $A_9 = \{0, 1,\ldots, 8\}$:

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

pools:

{0,3,6}

{1,4,7}

{2,5,8}

"layer" = partition of A_n

Shifted Transversal Design: idea

"Transversal" construction: layers.

"shift" variables from layer to layer

- limit co-occurrence of variables
- constant-sized intersection between pools

STD(n;q;k) : **n** variables, **q** prime, q < n, **k** number of layers (k ≤ q+1)

- First q layers: symmetric construction, q pools of size n/q or n/q+1
- If k=q+1: additional singular layer, up to q pools of heterogeneous sizes

Let:

- $\Gamma(q,n) = \min\{\gamma \mid q^{\gamma+1} \geq n\}$
- σ_q circular permutation on $\{0,1\}^q$: $\sigma_q \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_q \end{bmatrix} = \begin{bmatrix} x_q \\ x_1 \\ \vdots \\ x_{q-1} \end{bmatrix}$

STD Construction

$\forall$ j $\in$ {0,…,q}: Mj q$\times$n Boolean matrix, representing layer L(j)

columns $C_{j,0},\ldots,C_{j,n-1}$:

$$C_{0,0} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$, and $\forall$ i $\in$ {0,…,n-1} $C_{j,i} = \sigma_q^{\,s(i,j)}(C_{0,0})$ where:

- if j < q: $s(i,j) = \sum_{c=0}^{\Gamma} j^c \cdot \left\lfloor \frac{i}{q^c} \right\rfloor$
- if j=q (singular layer): $s(i,q) = \left\lfloor \frac{i}{q^\Gamma} \right\rfloor$

For k $\in$ {1,2,..., q+1}, STD(n;q;k) = $\bigcup_{j=0}^{k-1} L(j)$

STD example: n=9, q=3

$$M_0 = \begin{bmatrix} \boxed{1} & 0 & 0 & \boxed{1} & 0 & 0 & \boxed{1} & 0 & 0 \\ 0 & \boxed{1} & 0 & 0 & \boxed{1} & 0 & 0 & \boxed{1} & 0 \\ 0 & 0 & \boxed{1} & 0 & 0 & \boxed{1} & 0 & 0 & \boxed{1} \end{bmatrix}$$

L(0) = {{0,3,6}, {1,4,7}, {2,5,8}}

$$M_1 = \begin{bmatrix} \boxed{1} & 0 & 0 & 0 & 0 & \boxed{1} & 0 & \boxed{1} & 0 \\ 0 & \boxed{1} & 0 & \boxed{1} & 0 & 0 & 0 & 0 & \boxed{1} \\ 0 & 0 & \boxed{1} & 0 & \boxed{1} & 0 & \boxed{1} & 0 & 0 \end{bmatrix}$$

L(1) = {{0,5,7}, {1,3,8}, {2,4,6}}

$$M_2 = \begin{bmatrix} \boxed{1} & 0 & 0 & 0 & \boxed{1} & 0 & 0 & 0 & \boxed{1} \\ 0 & \boxed{1} & 0 & 0 & 0 & \boxed{1} & \boxed{1} & 0 & 0 \\ 0 & 0 & \boxed{1} & \boxed{1} & 0 & 0 & 0 & \boxed{1} & 0 \end{bmatrix}$$

L(2) = {{0,4,8}, {1,5,6}, {2,3,7}}

$$M_3 = \begin{bmatrix} \boxed{1} & \boxed{1} & \boxed{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \boxed{1} & \boxed{1} & \boxed{1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \boxed{1} & \boxed{1} & \boxed{1} \end{bmatrix}$$

L(3) = {{0,1,2}, {3,4,5}, {6,7,8}}

STD(n=9;q=3;k=2) = L(0) $\cup$ L(1).

STD example: n=9 to 27, q=3

n=9, q=3, third layer (j=2):

$$M_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad L(2) = \{\{0,4,8\}, \{1,5,6\}, \{2,3,7\}\}$$

n=27, q=3, j=2:

$+1$ $\quad +(1+j)$ $\quad +(1+j+j^2)$

$$M_2 = \left[\begin{array}{ccccccccc|ccccccccc|ccccccccc} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

Layout of the talk

- Biological context
- Definition of STD
- **Properties: a solution to the pooling problem**
- Behavior and efficiency
- Application: protein-protein interaction mapping

Co-occurrence of variables

$\forall$ k $\in$ {1,...,q+1}, $\forall$ i $\in$ {0,…,n-1}: $pools_k(i)$ = {p $\in$ STD(n;q;k) | $A_i \in$ p}

Theorem: (q prime). $\forall$ $i_1,i_2 \in$ {0,…,n-1},

$[i_1 \neq i_2] \Rightarrow [\mathrm{Card}(pools_{q+1}(i_1) \cap pools_{q+1}(i_2)) \leq \Gamma(q,n)]$.

(Idea of) proof: $\mathrm{Card}(pools_{q+1}(i_1) \cap pools_{q+1}(i_2)) = \mathrm{Card}\ \{j \in \{0,\ldots,q\}, C_{j,i_1} = C_{j,i_2}\}$.

However, for j < q:

$$C_{j,i_1} = C_{j,i_2} \iff s(i_1, j) \equiv s(i_2, j) \mod q \iff \sum_{c=0}^{\Gamma} j^c \cdot \left(\left\lfloor \frac{i_1}{q^c} \right\rfloor - \left\lfloor \frac{i_2}{q^c} \right\rfloor \right) \equiv 0 \mod q$$

Since q is prime, $\mathbb{Z}/q\mathbb{Z}$ is the field GF(q);

And since $i_1 \neq i_2$, there exists at least one $c \leq \Gamma$ such that $\left(\left\lfloor \frac{i_1}{q^c} \right\rfloor - \left\lfloor \frac{i_2}{q^c} \right\rfloor \right) \neq 0 \mod q$.

We therefore have a non-zero polynomial (in j) of degree at most Γ on GF(q).

If $C_{q,i_1} \neq C_{q,i_2}$: OK.

If $C_{q,i_1} = C_{q,i_2}$, coefficient of j^Γ in the polynomial is zero by definition of $s(i,q)$: OK.

Example: $n=9$, $q=3$ (hence $\Gamma=1$)

$L(0) = \{\{0,3,6\}, \{1,4,7\}, \{2,5,8\}\}$,

$L(1) = \{\{0,5,7\}, \{1,3,8\}, \{2,4,6\}\}$,

$L(2) = \{\{0,4,8\}, \{1,5,6\}, \{2,3,7\}\}$,

$L(3) = \{\{0,1,2\}, \{3,4,5\}, \{6,7,8\}\}$.

$pools_4(0) = \{\{0,3,6\}, \{0,5,7\}, \{0,4,8\}, \{0,1,2\}\}$.

0 appears exactly once ($\Gamma=1$) with each other variable.

A solution in the absence of noise

Corollary 1: If there are **at most t positive variables** in A_n and in the **absence of noise**: STD(n;q;k) is a solution, when choosing q prime such that $t \cdot \Gamma(q,n) \leq q$, and $k = t \cdot \Gamma + 1$.

(Idea of) proof: algorithm 1 correctly tags all variables.

Algorithm 1:

1. all the variables present in at least one negative pool are tagged negative
2. any variable present in at least one positive pool where all other variables have been tagged negative, is tagged positive

Example with n=9, q=3

Let t=1: by corollary 1, $k=t\cdot\Gamma+1=2$ layers are sufficient

Single positive variable: 8

{{0,3,6}, {1,4,7}, {2,5,8},

{0,5,7}, {1,3,8}, {2,4,6}}

Algorithm 1:

1. 4 negative pools show that 0, 1, …, 7 are negative;
2. 2 positive pools each show that 8 is positive (since 2, 5, 1 and 3 negative).

Note: if more than t variables are positive, all tags are still correct but some variables may not be tagged: they are “unresolved” (“ambiguous”).

Error-correction

Corollary 2: If there are **at most t positive variables** in A_n and **at most E observation errors**: STD(n;q;k) is a solution, when choosing q prime such that $t \cdot \Gamma(q,n) + 2 \cdot E \leq q$, and $k = t \cdot \Gamma + 2 \cdot E + 1$.

(Idea of) proof: algorithm 2 correctly tags all variables. Any contradictory observation is erroneous.

Algorithm 2:

1. all the variables present in at least E+1 negative pools are tagged negative
2. any variable present in at least E+1 positive pools where all other variables have been tagged negative, is tagged positive

Error-correction (2)

Errors can be false-positives or false negatives

Corollary 3: If there are **at most t positive variables** in A_n and **at most E false positive and E false negative observations**: STD(n;q;k) is a solution, when choosing q prime such that $t \cdot \Gamma(q,n)+2 \cdot E \leq q$, and $k=t \cdot \Gamma+2 \cdot E+1$.

(Idea of) proof: same algorithm as corollary 2.

Error-detection

If more than E errors: detection if

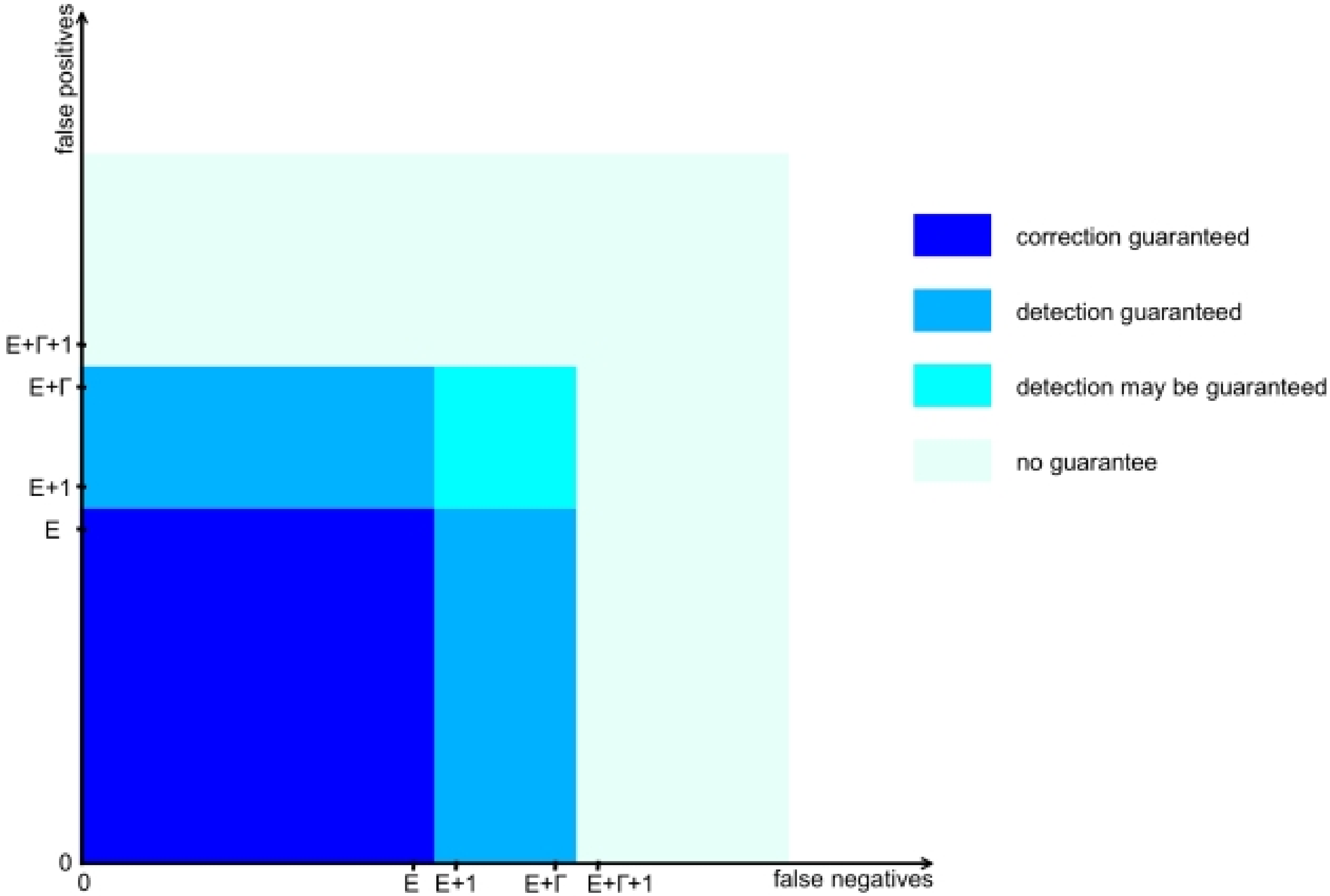
- some variables tagged twice or not at all
- more than t variables are tagged positive
- more than E observations identified as erroneous

Question: how many errors are necessary to avoid detection?

Answer:

- at least $E+\Gamma+1$ false negatives, or
- at least $E+\Gamma+1$ false positives, or
- if $E < 2 \cdot \Gamma - 1$: at least $3 \cdot E+2$ errors including at least $E+1$ errors of each type.

Error detection and correction



Even redistribution of variables

Theorem: Let m ≤ k ≤ q and consider $\{P_1,\ldots,P_m\} \subset \mathrm{STD}(n;q;k)$, each belonging to a different layer. Then:

$$\lambda_m \leq \left| \bigcap_{h=1}^{m} P_h \right| \leq \lambda_m + 1 \text{, where} \quad \lambda_m = \sum_{c=m}^{\Gamma} \left[\left\lfloor \frac{n-1}{q^c} \right\rfloor \% \, q \right] \cdot q^{c-m} \quad .$$

Proof: see BMC Bioinformatics 2006, 7:28.

Notes:

- λ_m depends only on m, not on the choice of the pools $P_1,\ldots,P_m$. Hence the theorem expresses that every pool, and every intersection between 2 or more pools, is redistributed evenly in each remaining layer
- L(q) does not work (k ≤ q)

Layout of the talk

- Biological context
- Definition of STD
- Properties
- **Behavior and efficiency**
- Application: protein-protein interaction mapping

Guaranteed efficiency

Problem specification (n, t, E) → minimal STD design

Example: n=10000, t=5, E=0

q	Γ (compression)	k (nb layers)	q·k (nb pools)
≤13	≥3	≥16	k>q+1
17	3	16	272
19	3	16	304
23	2	11	253
29	2	11	319
…	2	11	…
97	2	11	1067
101	1	6	606

Comparing with other designs

Comparing with other designs

- (1) optimal solution for some instances with $t \leq 2$. (2): real application with t=2 and n=1530; design with 4368 variables similar to (1) (but not optimal), reduced to 1530 variables to fit the problem spec. Finally: **similar number of pools** and pool size as STD.

1. Balding D., Torney D. (1996) *J. Comb. Theory Ser A* **74**, 131-140.
2. Balding D., Torney D. (1997) *Fungal genet. biol.* **21**, 302-307.

Comparing with other designs

- (1) optimal solution for some instances with $t \leq 2$. (2): real application with t=2 and n=1530; design with 4368 variables similar to (1) (but not optimal), reduced to 1530 variables to fit the problem spec. Finally: **similar number of pools** and pool size as STD.
- (3,4) designs guaranteeing t=2 often work well for larger t. Example $n=10^6$: **v=946** pools => guarantee for **t=2** and 97.1% success for **t=5**. STD(n;11;11): **v=121, t=2**; STD(n;23;21): **v=483, t=5** (guaranteed).

1. Balding D., Torney D. (1996) *J. Comb. Theory Ser A* **74**, 131-140.
2. Balding D., Torney D. (1997) *Fungal genet. biol.* **21**, 302-307.
3. Macula A. (1996) *Discrete Math.* **162**, no. 1-3, 311-312.
4. Macula A. (1999) *Ann. Comb.* **3**, 61-69.
5. Ngo H., Du D-Z. (2002) *Discrete Math.* **243**, no. 1-3, 161-170.

Comparing with other designs

- (1) optimal solution for some instances with t ≤ 2. (2): real application with t=2 and n=1530; design with 4368 variables similar to (1) (but not optimal), reduced to 1530 variables to fit the problem spec. Finally: **similar number of pools** and pool size as STD.
- (3,4) designs guaranteeing t=2 often work well for larger t. Example n=10^6: **v=946** pools => guarantee for **t=2** and 97.1% success for **t=5**.

  STD(n;11;11): **v=121, t=2**; STD(n;23;21): **v=483, t=5** (guaranteed).
- (5) two constructions (graph theory). Example n=18 918 900: **v=5460** pools => guarantee for **t=2**, and 98.5% success for **t=9**. STD(n;13;13): **v=169, t=2**; STD(n;37;37): **v=1369, t=9** guaranteed.

1. Balding D., Torney D. (1996) *J. Comb. Theory Ser A* **74**, 131-140.
2. Balding D., Torney D. (1997) *Fungal genet. biol.* **21**, 302-307.
3. Macula A. (1996) *Discrete Math.* **162**, no. 1-3, 311-312.
4. Macula A. (1999) *Ann. Comb.* **3**, 61-69.
5. Ngo H., Du D-Z. (2002) *Discrete Math.* **243**, no. 1-3, 161-170.

Layout of the talk

- Biological context
- Definition of STD
- Properties
- Behavior and efficiency
- **Application: protein-protein interaction mapping**

Using STD

- In practice, if we **tolerate** a small fraction of **ambiguous variables**, we can use less pools than necessary for the guarantee

Example: n=10000, t=5, error-rate 1%: guarantee requires 483 pools; but when tolerating up to 10 ambiguous variables (will need retesting), 143 pools prove sufficient

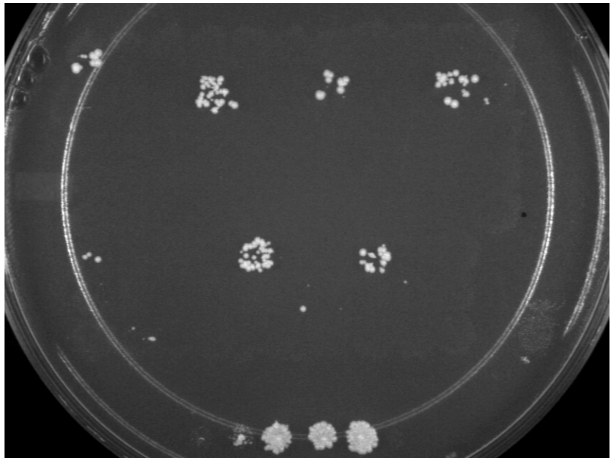
- Given (n,t,E) and number of tolerated ambiguous variables, we find optimal parameter values by simulation
- Difficulty: “decode” observed pool values

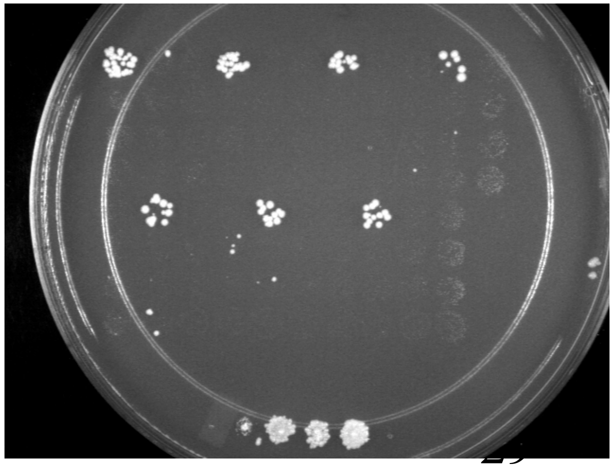
For this purpose, new algorithms (paper in prep.)

Example: Y2H pilot project

Collaboration with Marc Vidal's lab, DFCI, Boston

- **n=940** preys from human ORFeome
- noise levels unknown, estimated at 20% false negatives and 20% false positives
- combined into **169 pools** of 73 preys, 13x redundancy (2 days of work with robot)
- 100 baits screened; the 100x940 pairs have all been tested previously
- Initial results:
  - ➢ 38 known interactions (72%)
  - ➢ 23 new interactions (improved twofold)
  - ➢ better estimates for error-rates





Summary: the Shifted Transversal Design

- Family of non-adaptive combinatorial pooling designs
- Solution to the "pooling problem"
- Flexibility: for any (n,t,E), guarantee requirement satisfied
- Efficiency: STD seems more efficient than most published pooling designs
- Applied to protein-protein interaction mapping, successful

Prospects

- Study STD from the point of view of Shannon's information theory (are we far from the theoretical optimum?)

- Smart-pools for the full *C. elegans* ORFeome: desire for a modular construction

build once, use with various pool sizes (assay in 96, 384, 1536, 6144…)

STD seems well suited for this!

Example: n=27, q=3

$$M_2 = \left[\begin{array}{ccccccccc|ccccccccc|ccccccccc} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

Acknowledgments

M. Vidal, J.-F. Rual, D. Hill. Dana Farber Cancer Institute, Boston

L. Trilling, J.-L. Roch. IMAG Institute, Grenoble

Funding: Institut National Polytechnique de Grenoble

A. Duda (LSR-IMAG, Grenoble)

Thierry-Mieg N. A new pooling strategy for high-throughput screening: the Shifted Transversal Design. BMC Bioinformatics 2006, 7:28.

Thierry-Mieg N. Pooling in systems biology becomes smart. Nat Methods. 2006; 3(3):161-2.