CanaDAM 2013
June 10 - 13
Memorial University of Newfoundland

Michael Braun (michael.braun@h-da.de)

University of Applied Sciences, Darmstadt, Germany
joint work with Jan Reichelt

q-analogs of packing designs

$S_q(t, k, n)$ q-Steiner system

=

set of k-subspaces of $GF(q)^n$

such that

each t-subspace of $GF(q)^n$

is contained in

exactly one

of the selected k-subspaces

Example: $S_2(2, 3, 13)$ does exist

research problem:
do $S_2(2, 3, 7)$ q-Steiner systems exist?

asked by Thomas in 1987

it would have size 381
(packing bound = size of q-Steiner system)

new research problem:
what is the size of a maximal packing?

→ q-packing design

$P_q(t, k, n)$ q-packing design

~~$S_q(t, k, n)$ q-Steiner system~~

=

set of k-subspaces of $GF(q)^n$

such that

each t-subspace of $GF(q)^n$

is contained in

~~exactly~~ one

of the selected k-subspaces

at most

bounds on size for $P_2(2, 3, n)$

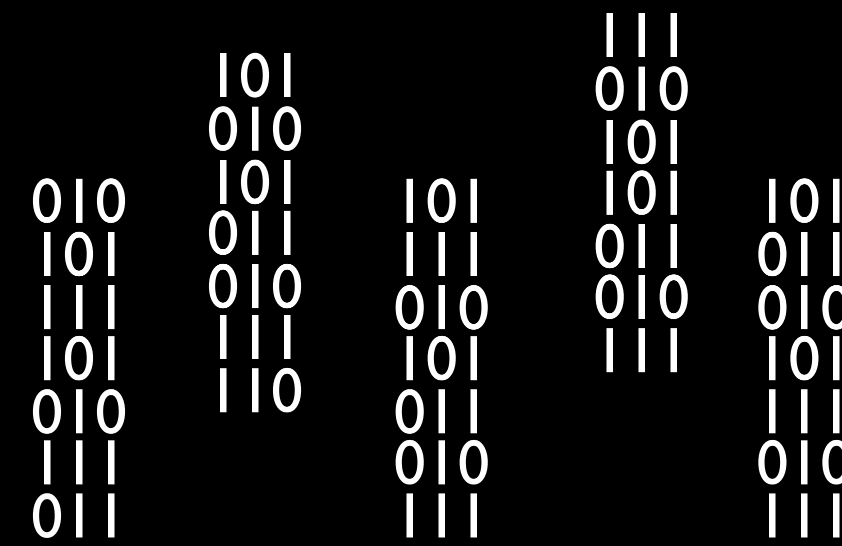
n	lower (by construction)	upper (packing bound)	reference
6	77	93	Kohnert, Kurz, 2008
7	304	381	K, K
8	1275	1542	K, K
9	5694	6205	Etzion, Vardy, 2008
10	21483	24893	K, K
11	79833	99718	K, K
12	315315	399165	K, K
13	1597245	1597245	B, E, Östergård, V, W
14	4770411	6390150	E, Silberstein, 2013

application:
q-packing designs correspond to random network codes

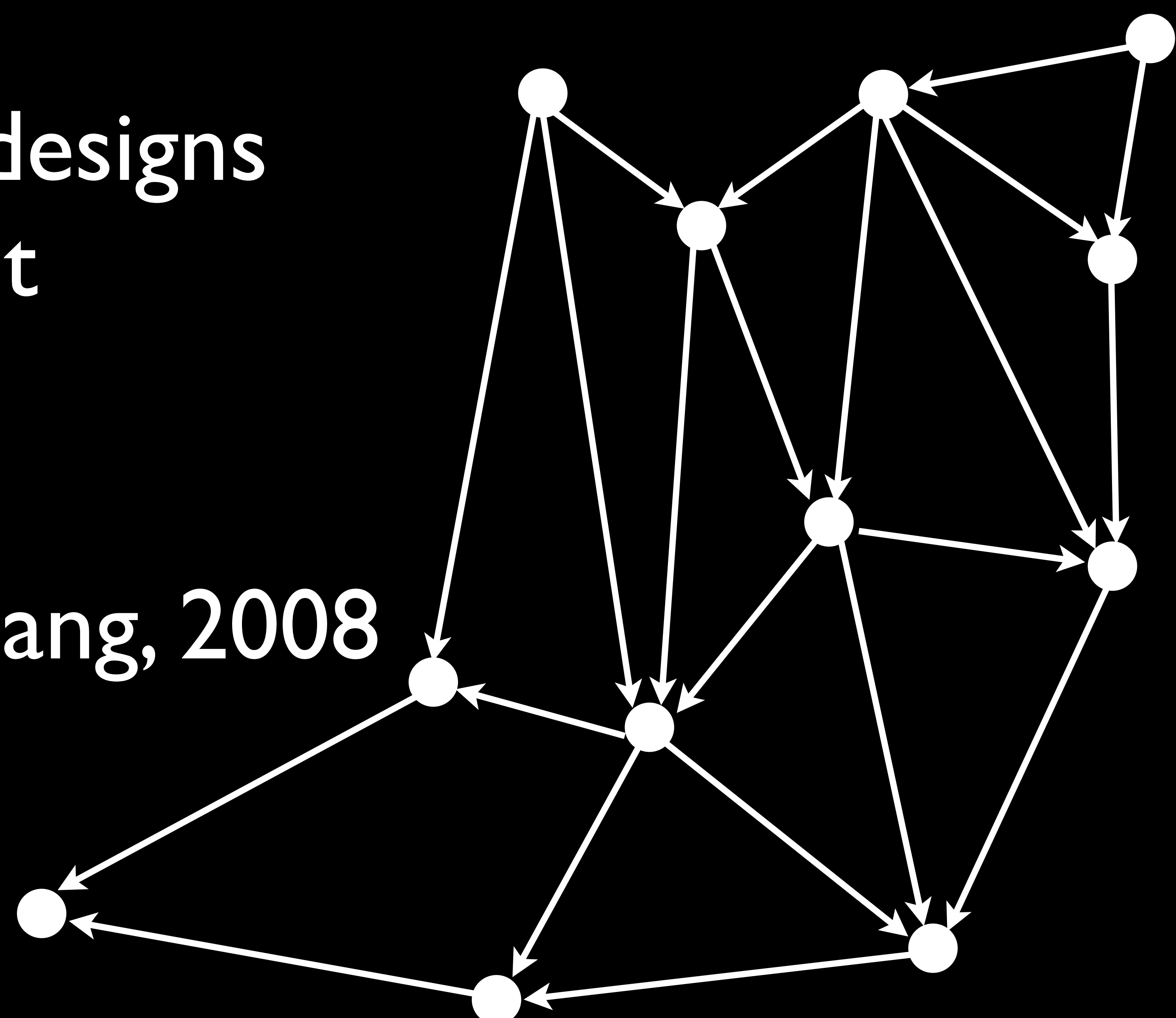
constant dimension code

=

set of k-subspaces of $GF(q)^n$

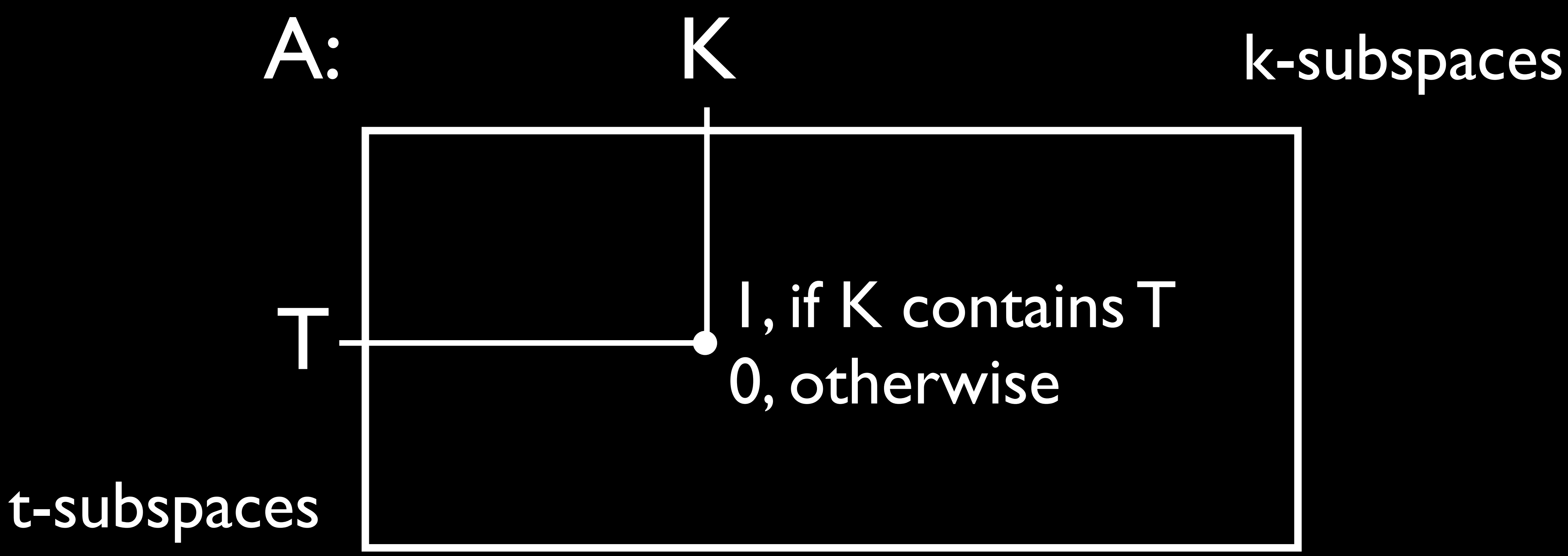


maximal q-packing designs are optimal constant dimension codes



Koetter & Kschischang, 2008

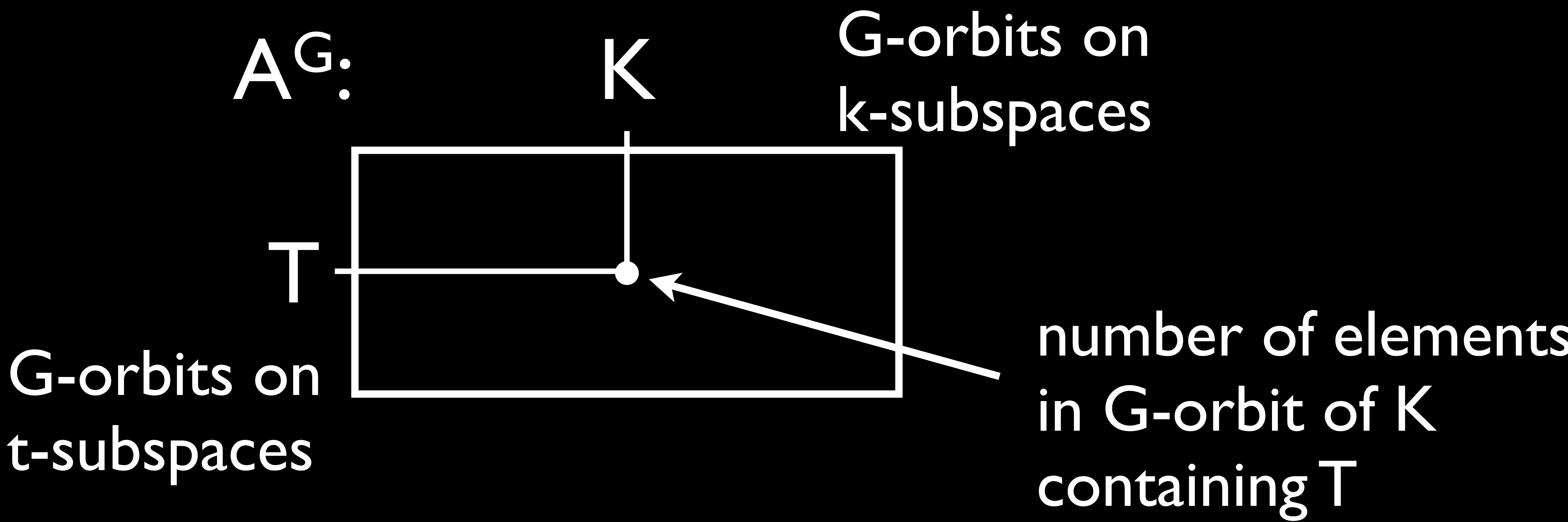
construction of
maximal q-packing design
is integer linear programming



find 0-1 vector x with
$Ax \leq 1$ and such that
weight of x is maximal

in general,
this optimization problem
is unfortunately too large

so, let groups $G \leq GL(n, q)$
do the work to reduce the problem



find 0-1 vector x with

$A^G x \leq 1$ and such that

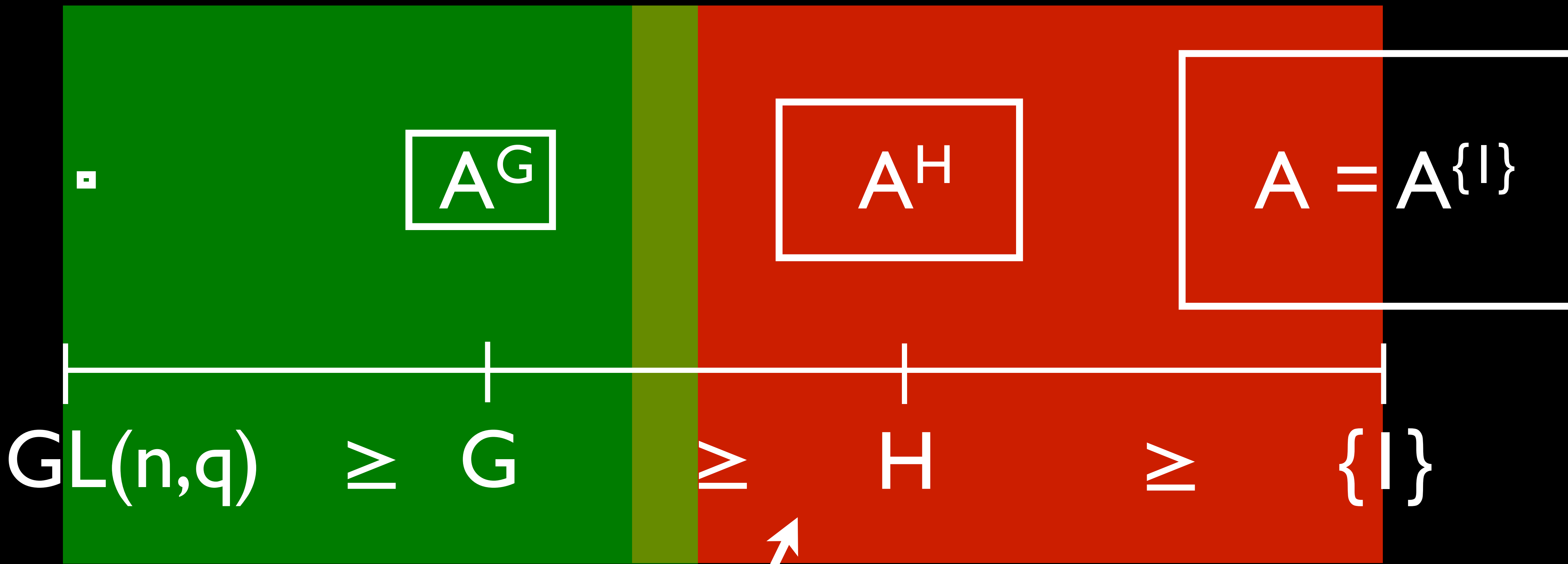
sum of sizes of selected orbits is maximal

idea goes back to Kramer & Mesner, 1976
for the construction of ...

... combinatorial t-(n, k, λ) designs
$\Rightarrow A^G x = \lambda, G \leq Sym(n)$

... q-analogs of t-(n, k, λ) designs
$\Rightarrow A^G x = \lambda, G \leq GL(n,q)$

the approach seems to be nice but it has some limits ...



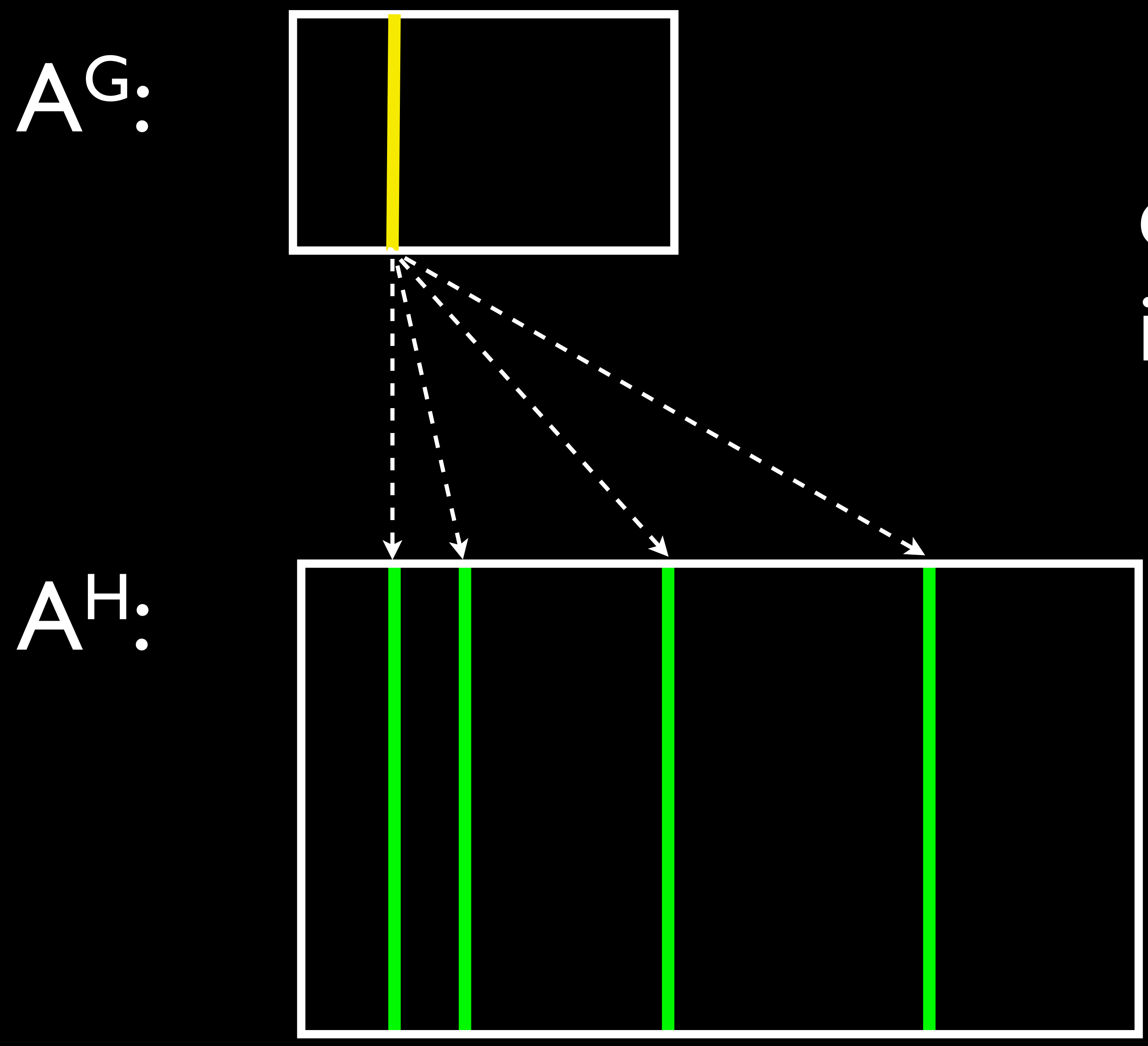
computationally feasible

computationally unfeasible

objects admitting small groups of automorphisms have to be searched in the *red* area what, if we want to search in the *red* area?

do descended Kramer-Mesner: start with big groups G and finish with subgroups $H \leq G$

G-orbits split into
H-orbits for H $\leq$ G



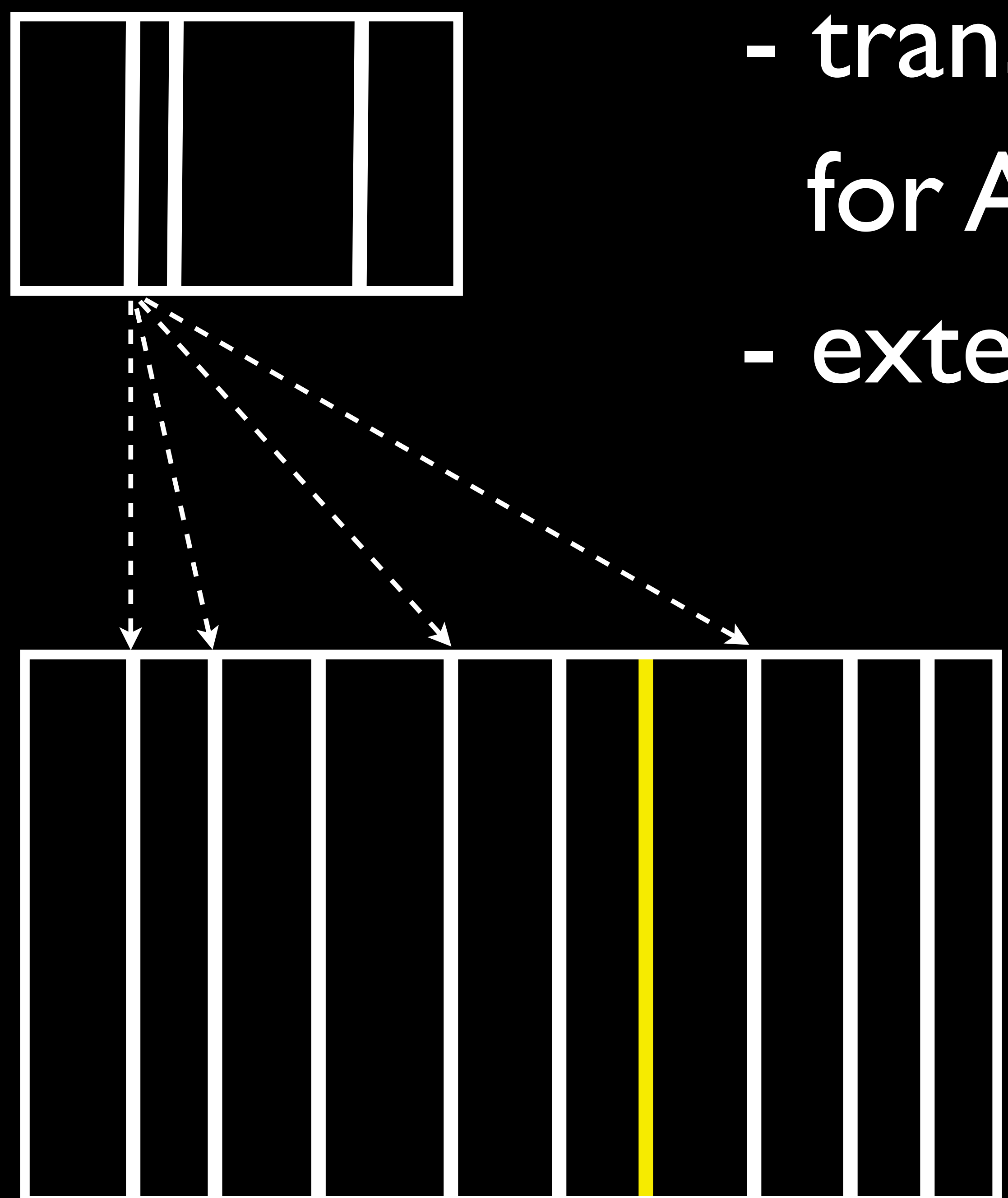
columns of A^G split into columns of A^H

solutions x of $A^G x \leq 1$ split into solutions y of $A^H y \leq 1$

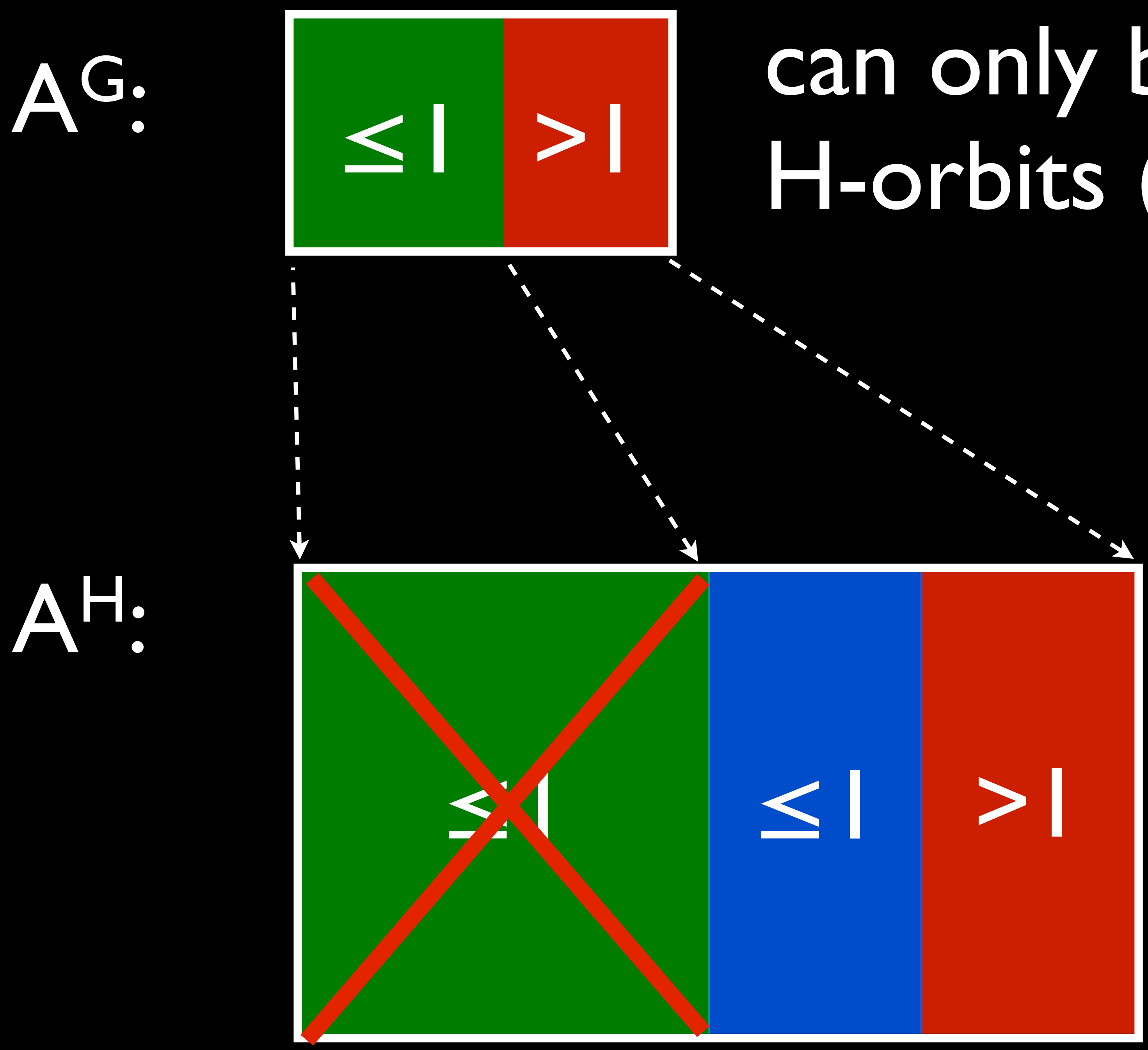
- find solution x for A^G
- transform x into solution for A^H
- extend solution for A^H

A^G:

A^H:



maximal solutions x for A^G can only be extended by *blue* H-orbits (not by *green* ones)



...hence, it can be advantageous to work with non-maximal solutions

some results

new bounds on size for $P_2(2, 3, n)$

n	new	old	packing
7	329	304	381
8	1312	1275	1542
11	92411	79833	99718
12	385515	315315	399165
14	5996178	4770411	6390150

the approach could be appropriate
for further discrete structures

blocking sets

packing designs

caps

covering designs

linear codes

partial spreads

q-covering designs

arcs

etc?

thank you very much
for your attention