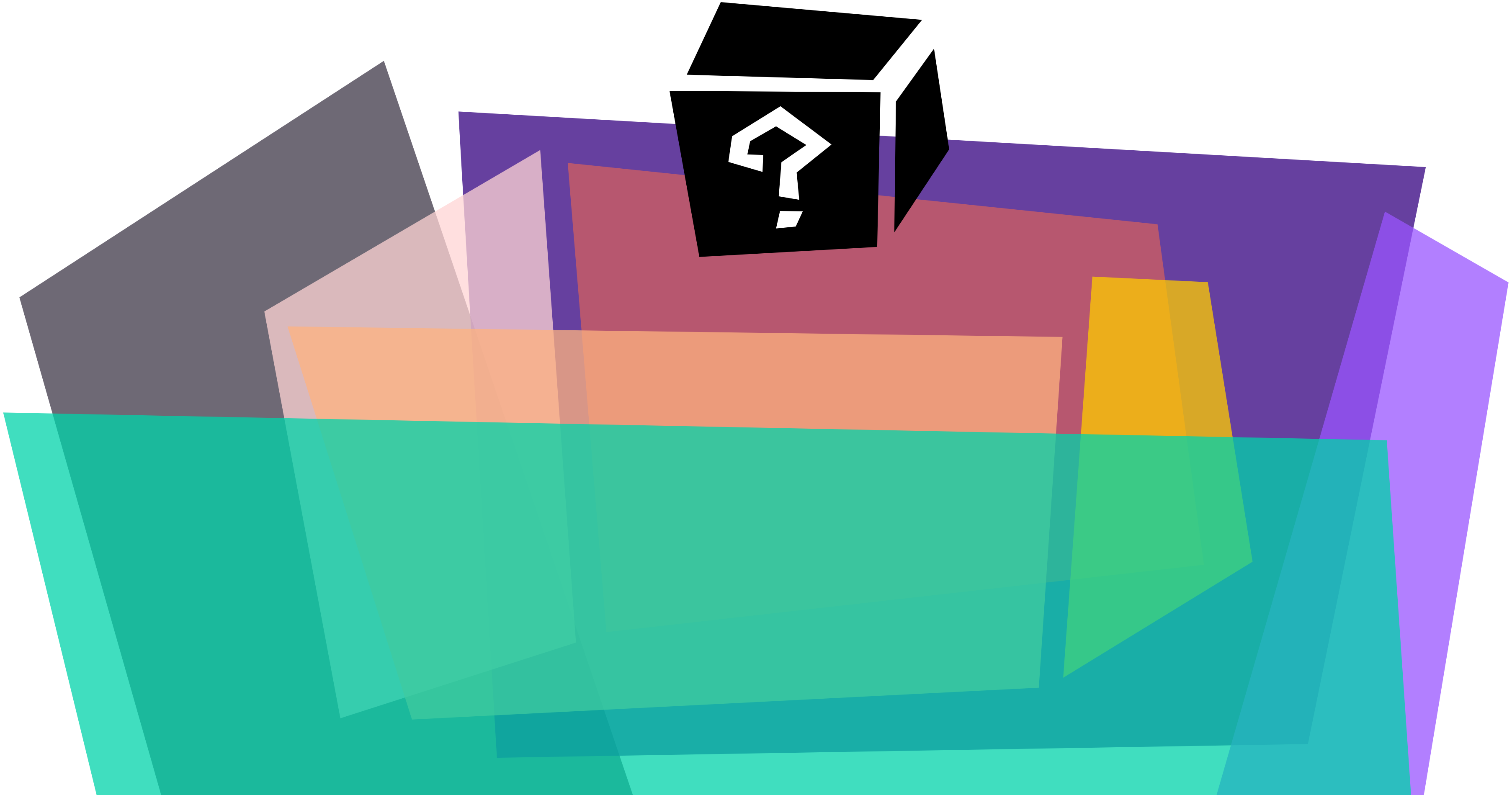
Nullable Compositions

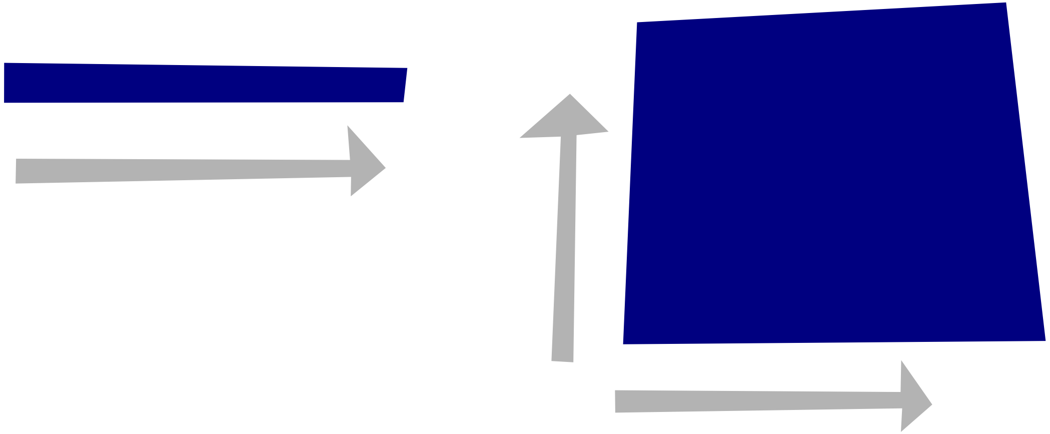
disjoint work with

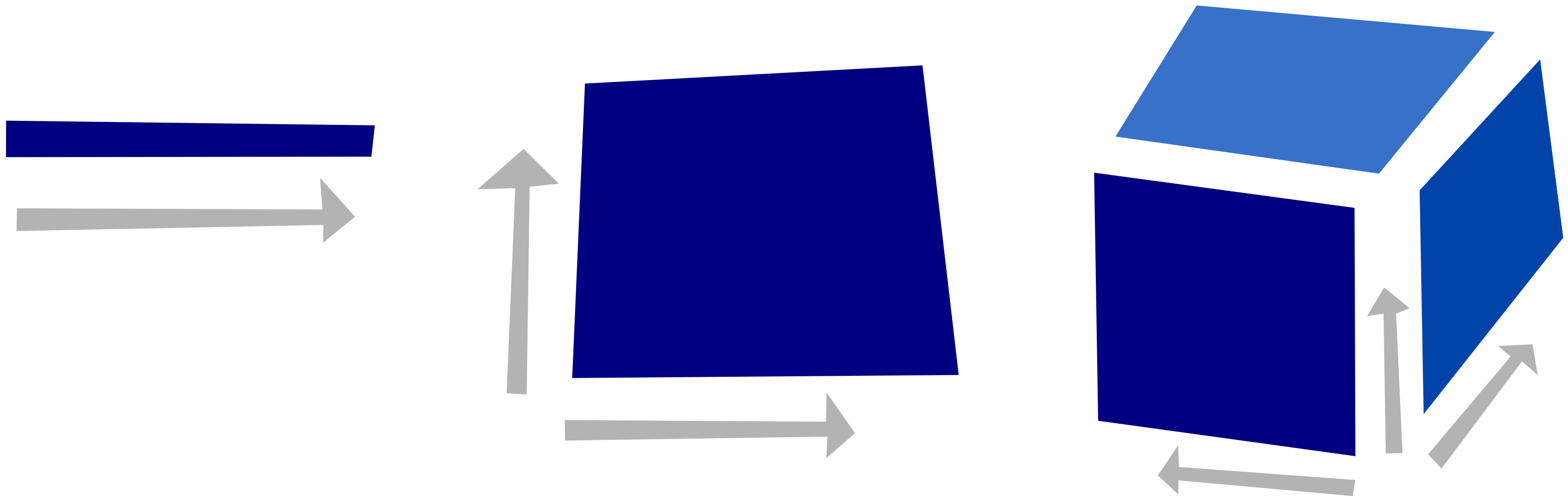
Evan Cavallo / Carlo Angiuli
Anders Mörtberg / Andrea Vezzosi

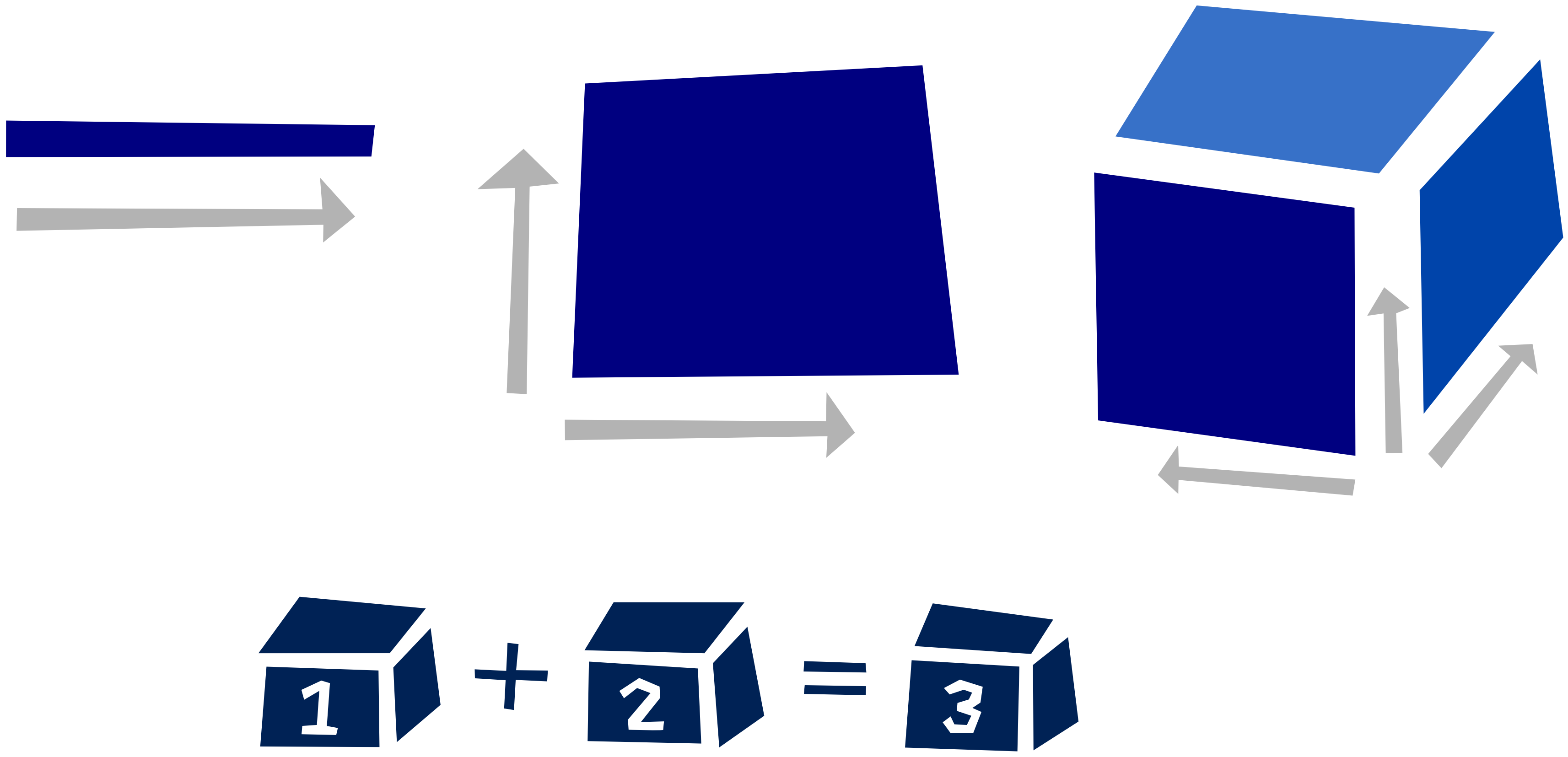
ICMS
15 July 2020

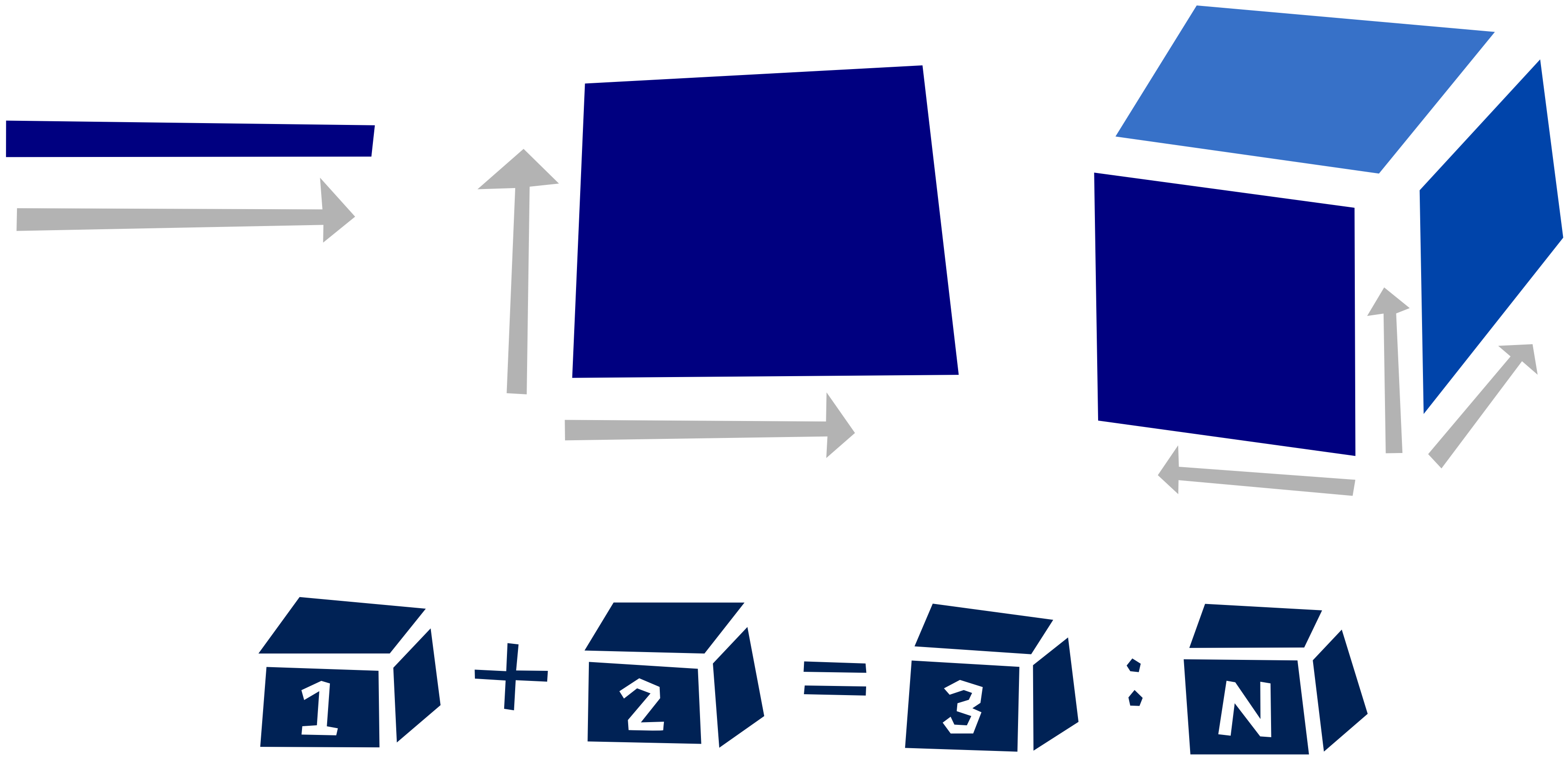
Favonia

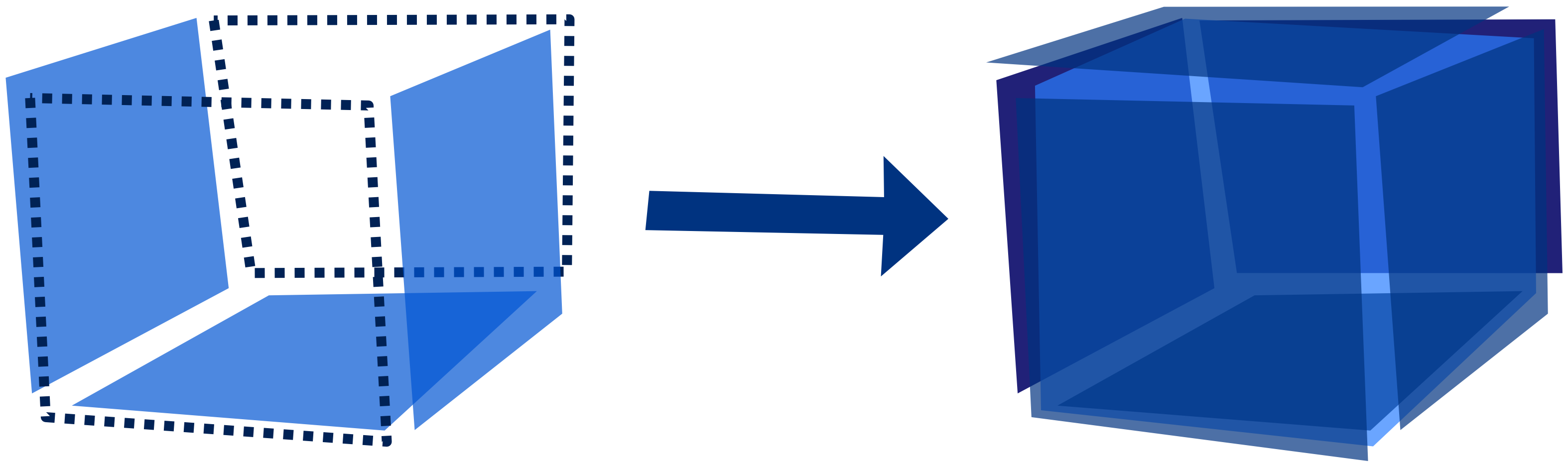


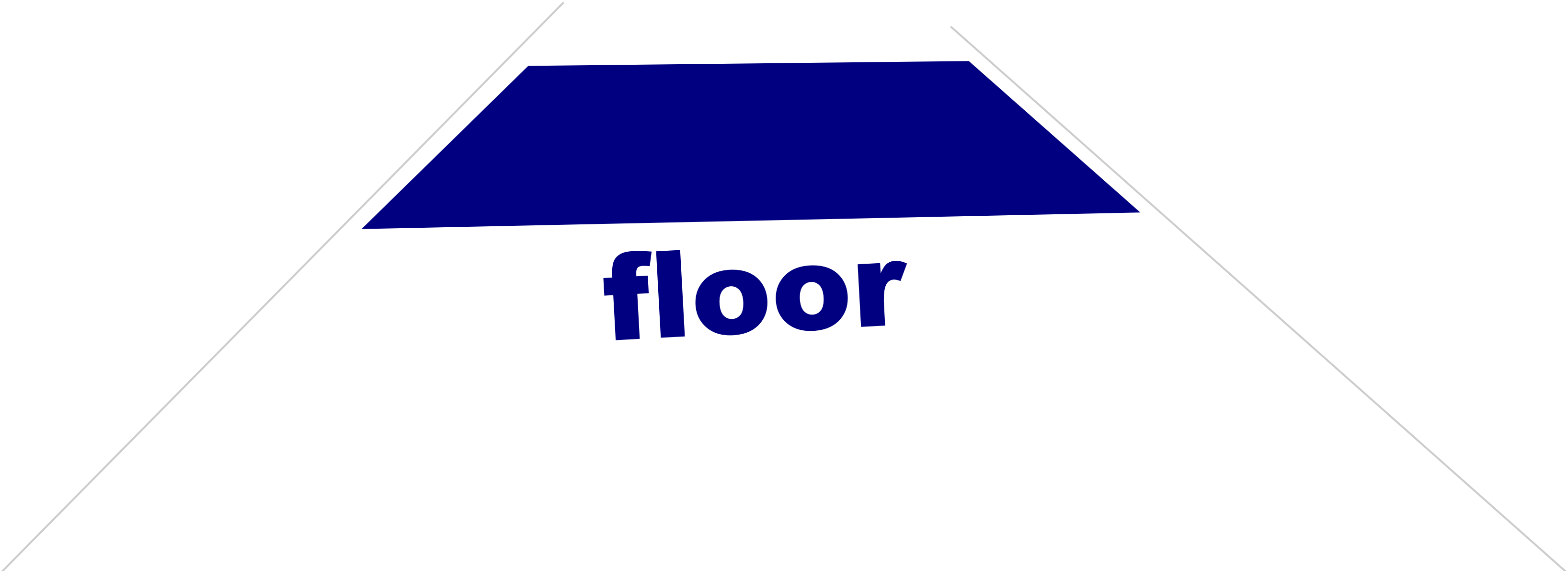


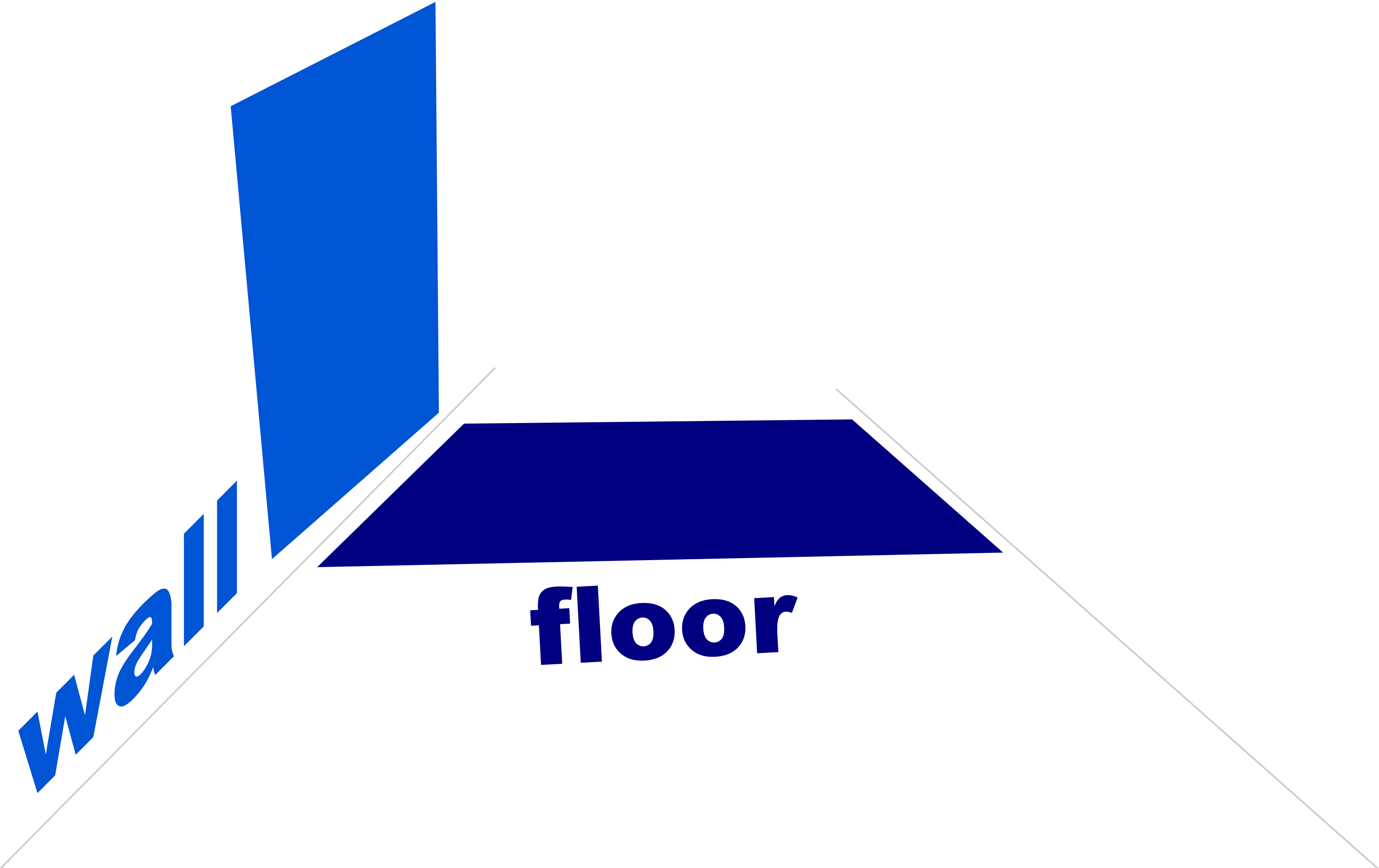


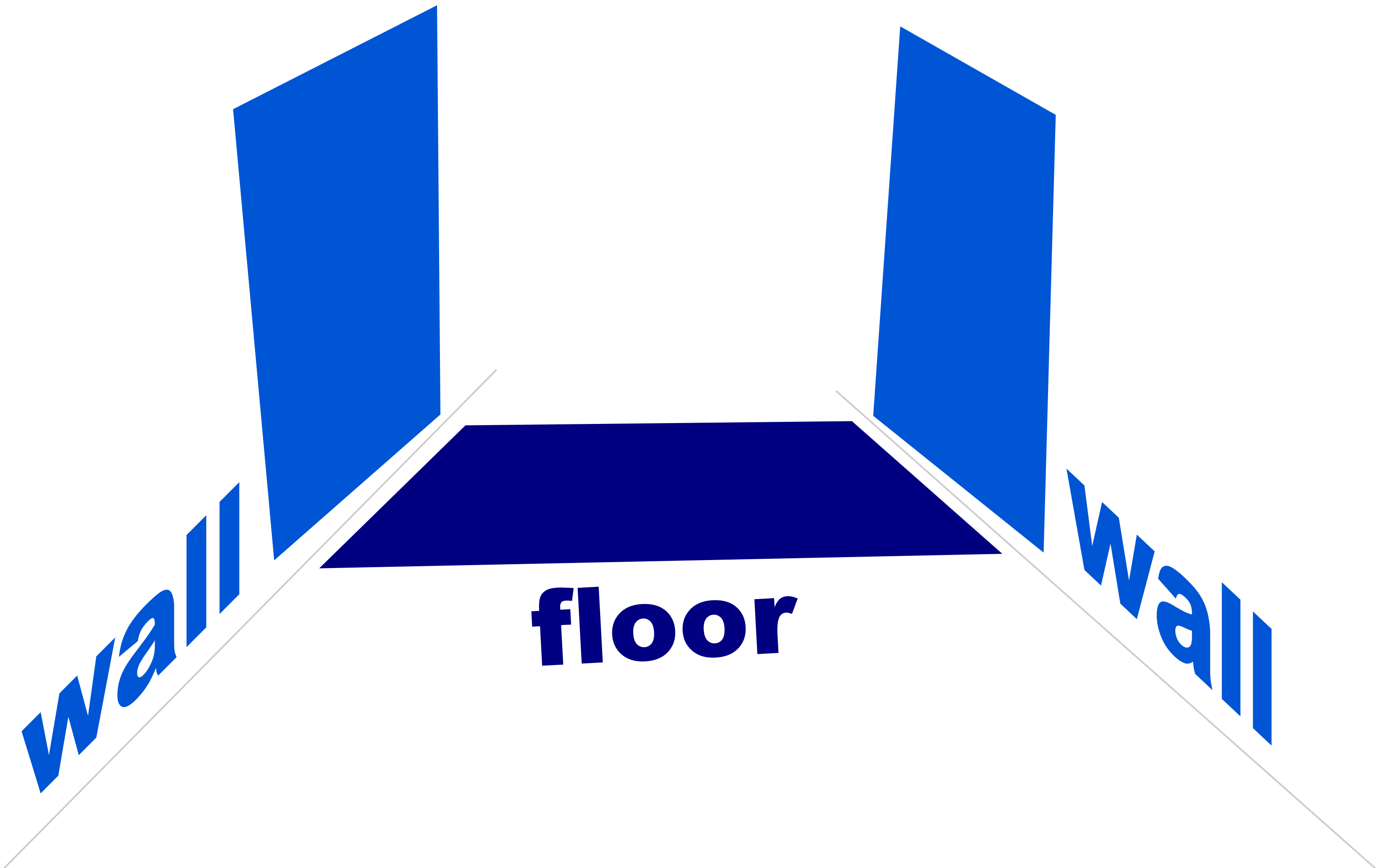

1 + 2 = 3

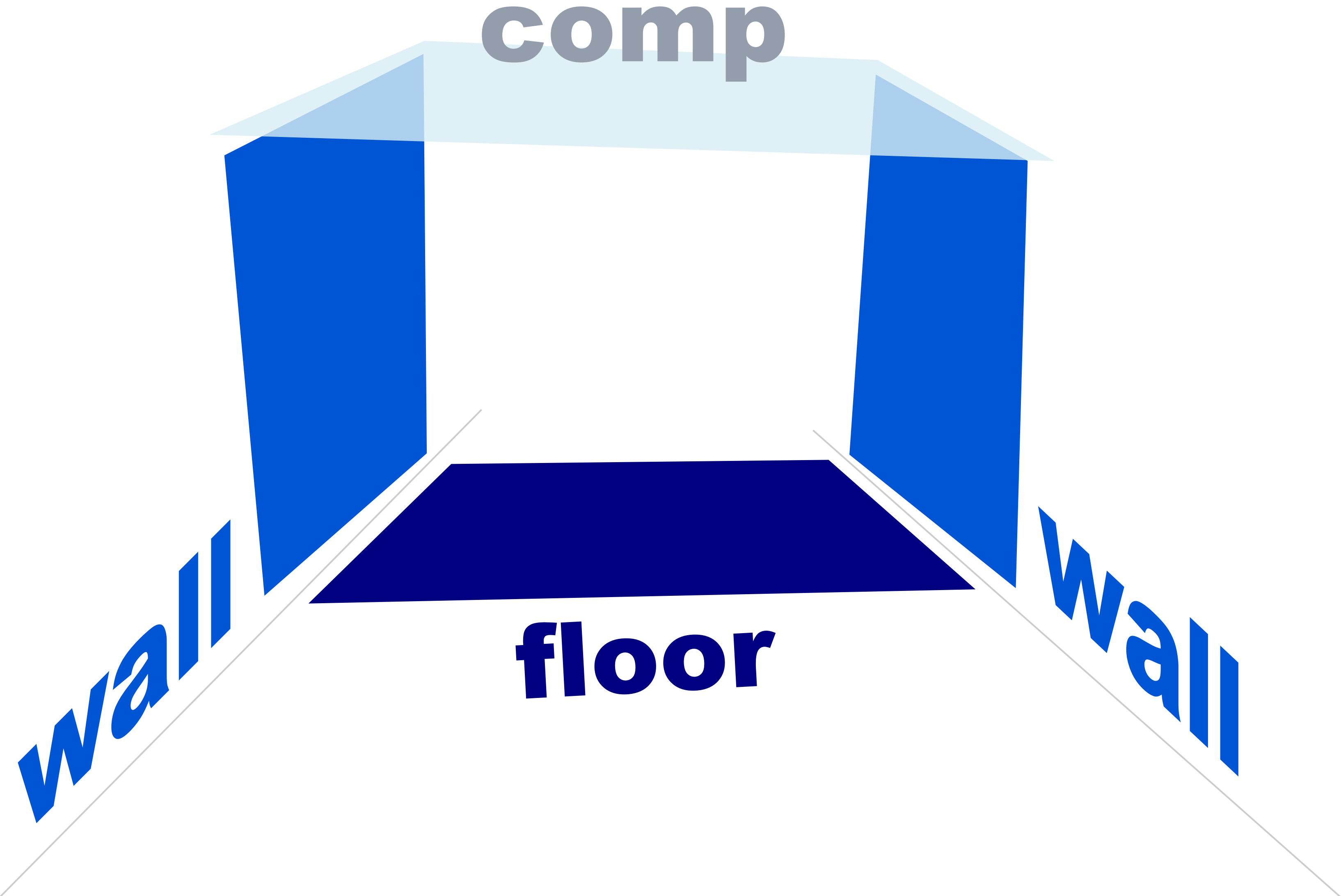

1 + 2 = 3 : N




floor


wall
floor


wall
floor
wall


comp
wall
floor
wall

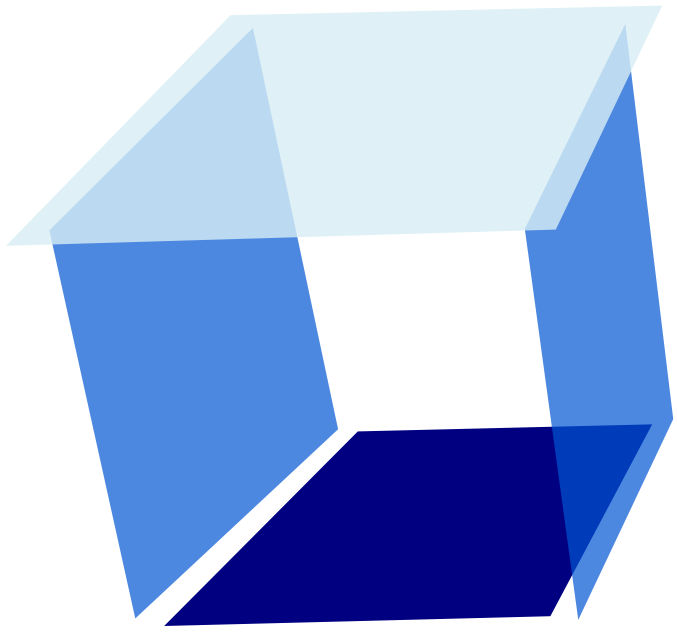
cubes

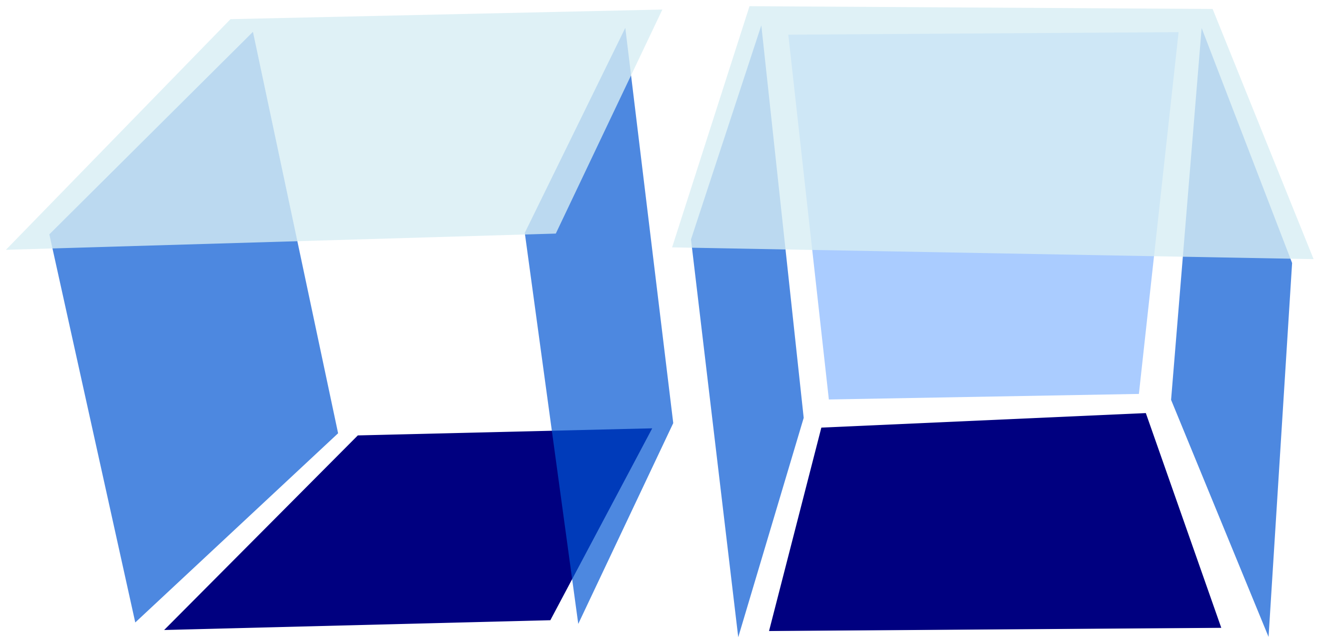
+) composition

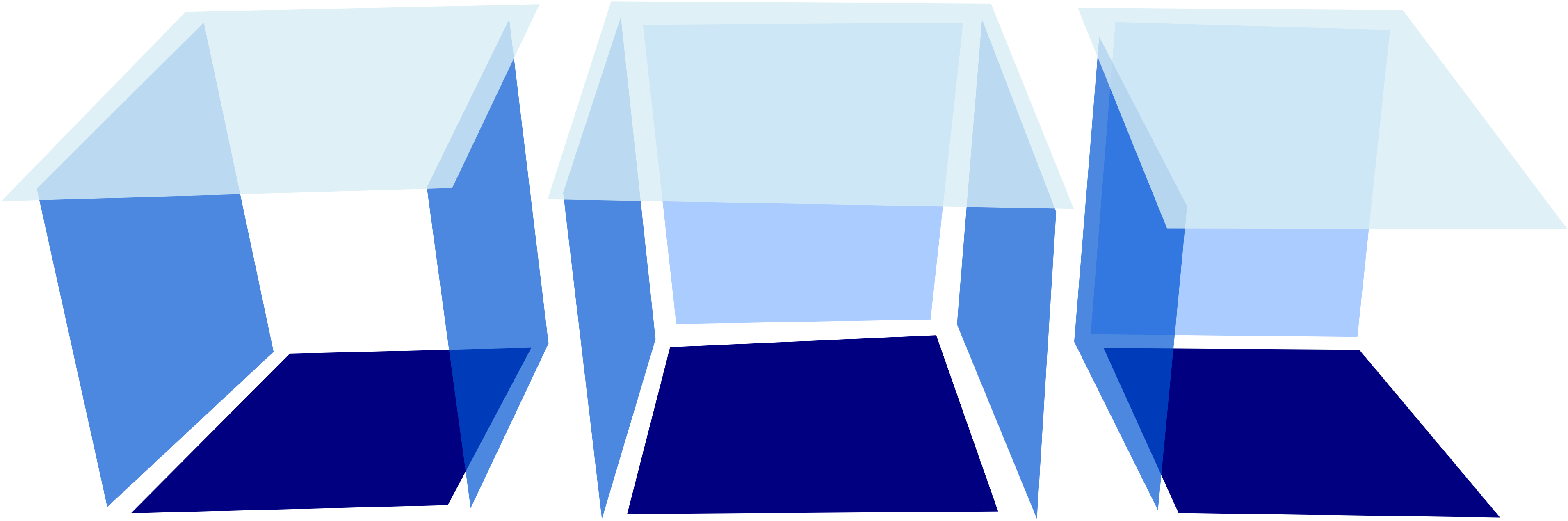
---

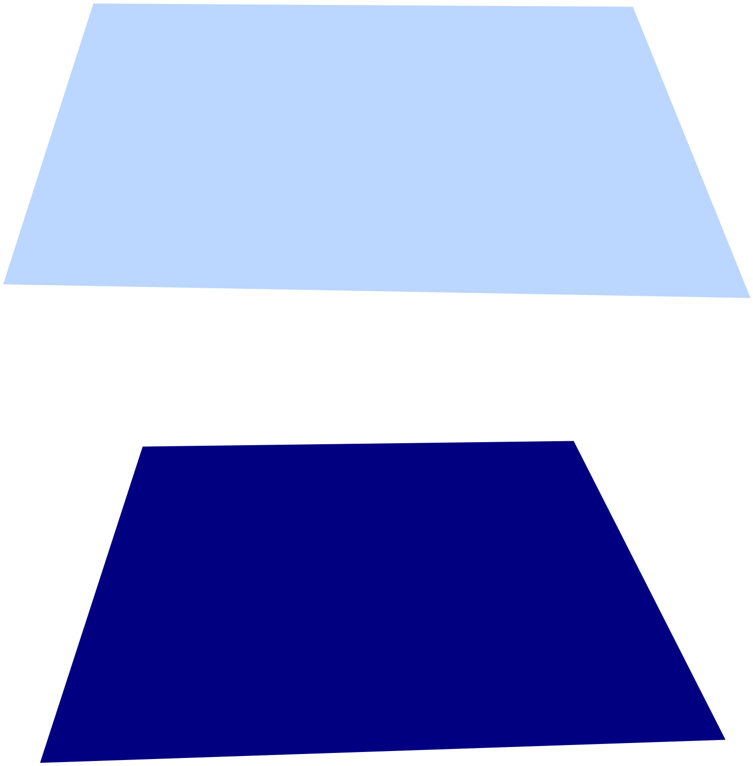
cubical TT

major difficulty: composition for univalent universes









null compositions = no walls

Brunerie's number

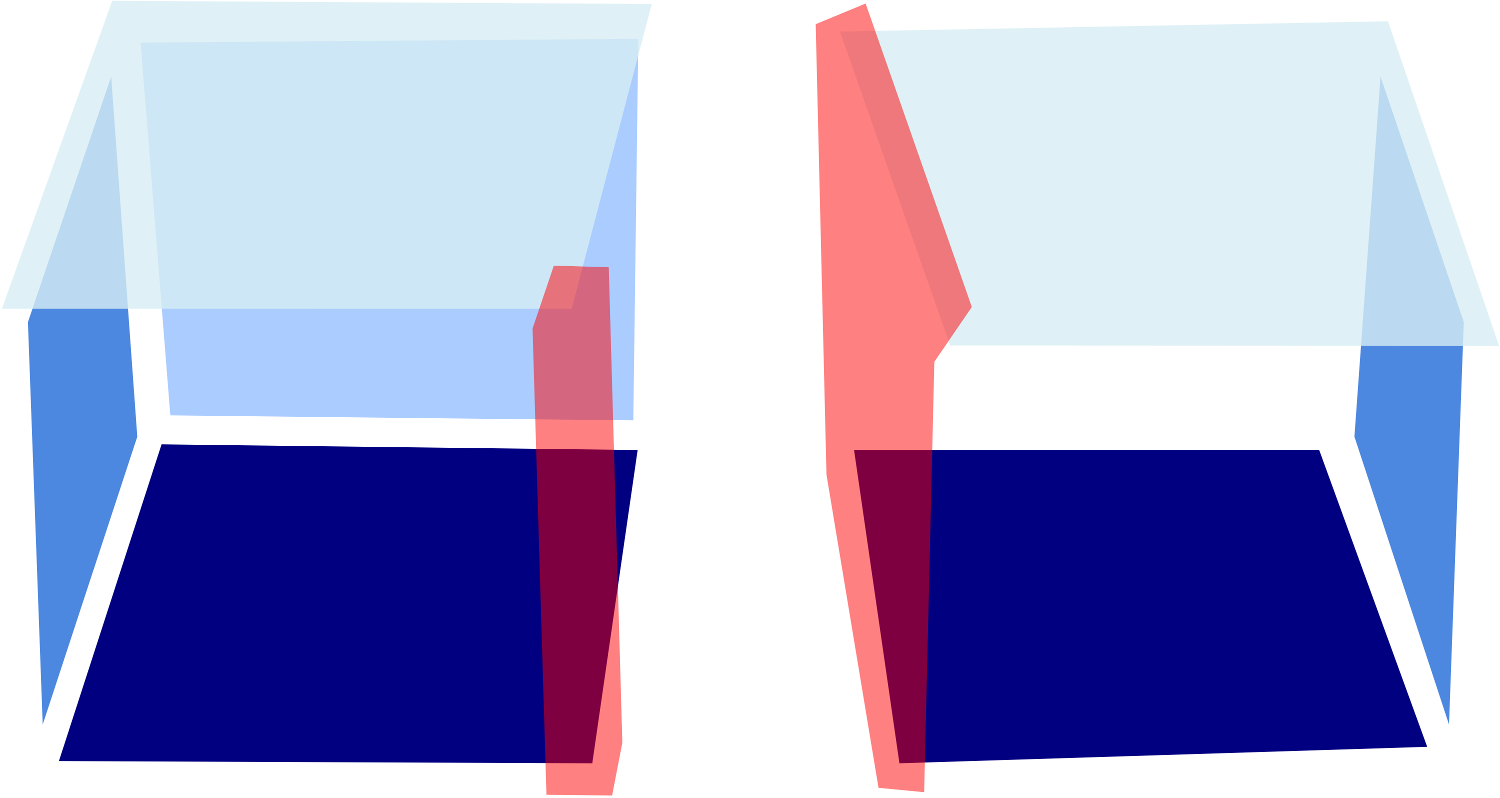
a program that should output 2*

$$\pi_4(S^3) \simeq \mathbb{Z}/n\mathbb{Z}$$

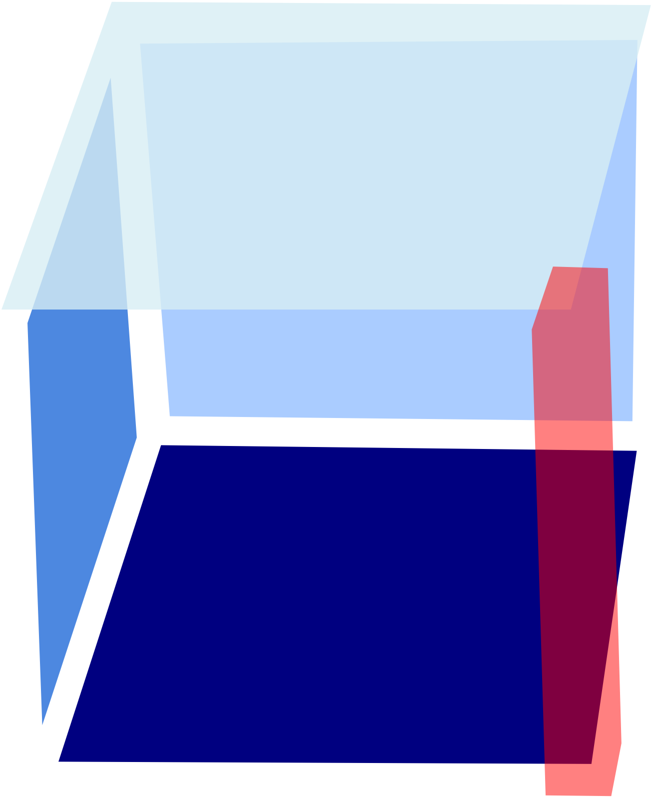
*read Guillaume Brunerie's thesis

```
(p = (<i> <j> <k> ((test0104 @ j) @ k) @ i)))))
i = 1))))))))))))))))))))))))))))))))))))))))))))
))) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
[]) []) []) []) []) []) []) []) []) []) []) [])
```

2019.03.04-cubicaltt-fbdb422ada0287dbfc7b097c4a9355ed501be6e6-stack-lts9.5-brunerie2-brunerie_opt-2.output.gz



nullable compositions



nullable

not covering every corner

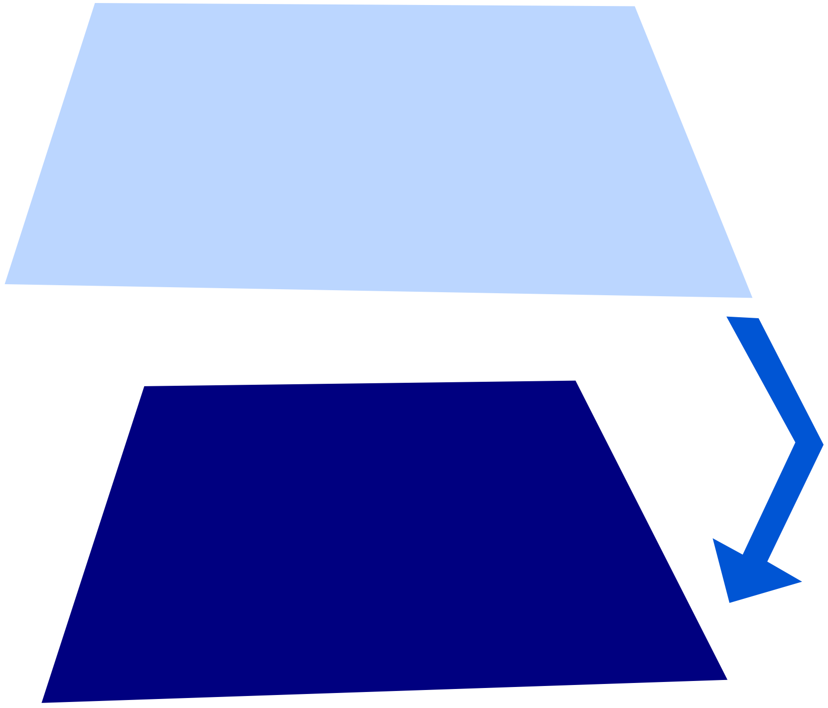
not "true" under double negation

not "true" under some closed substitutions

kill nullable compositions!

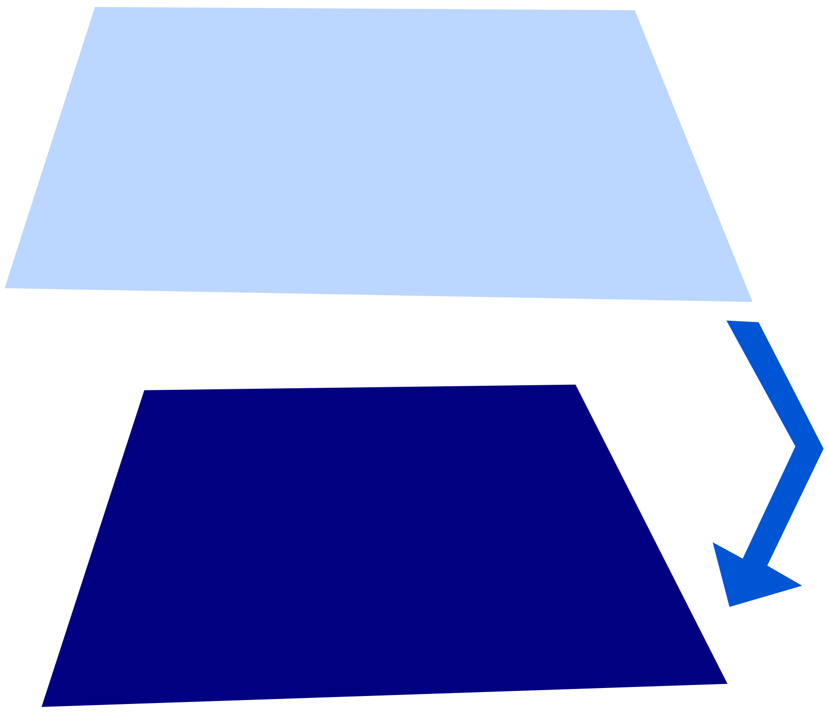
HOW?

Plan A



reduces to floor if null?

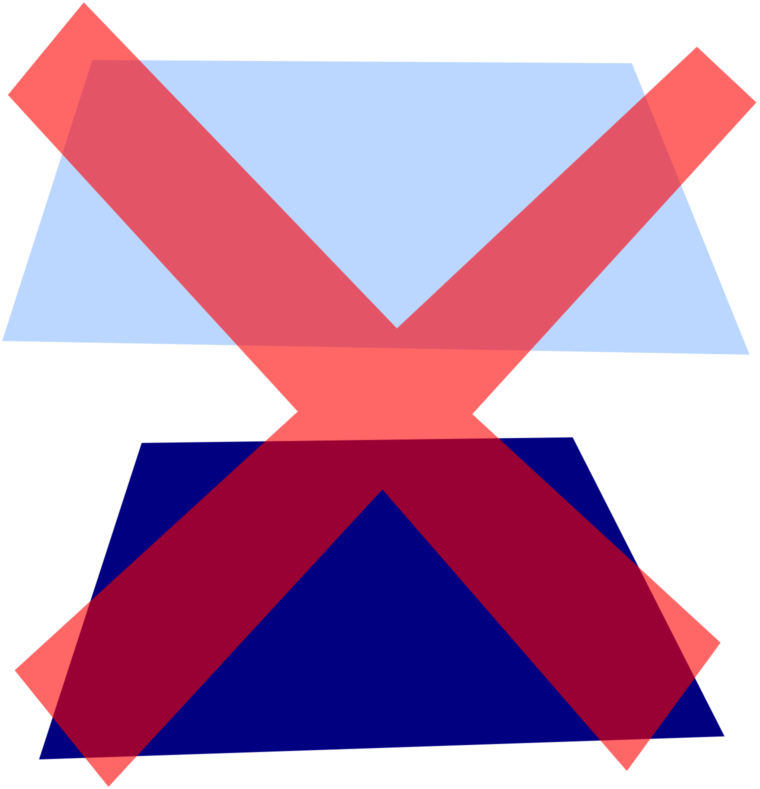
Plan A



reduces to floor if null?

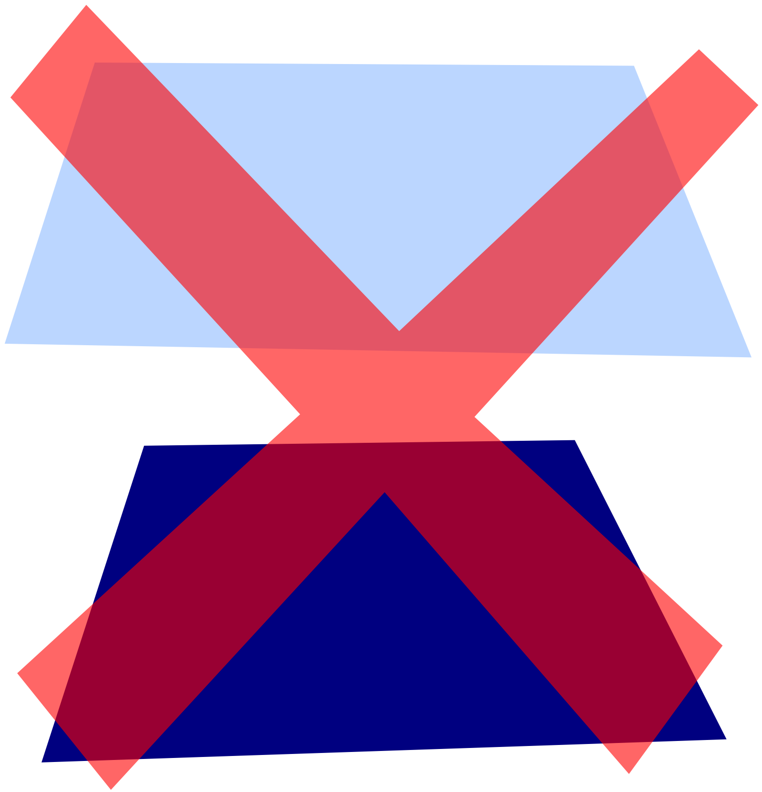
difficult with univalence

keyword: regularity



Plan B

ban nullable compositions?



Plan B

ban nullable compositions?

but universes need them

in current constructions

Plan C

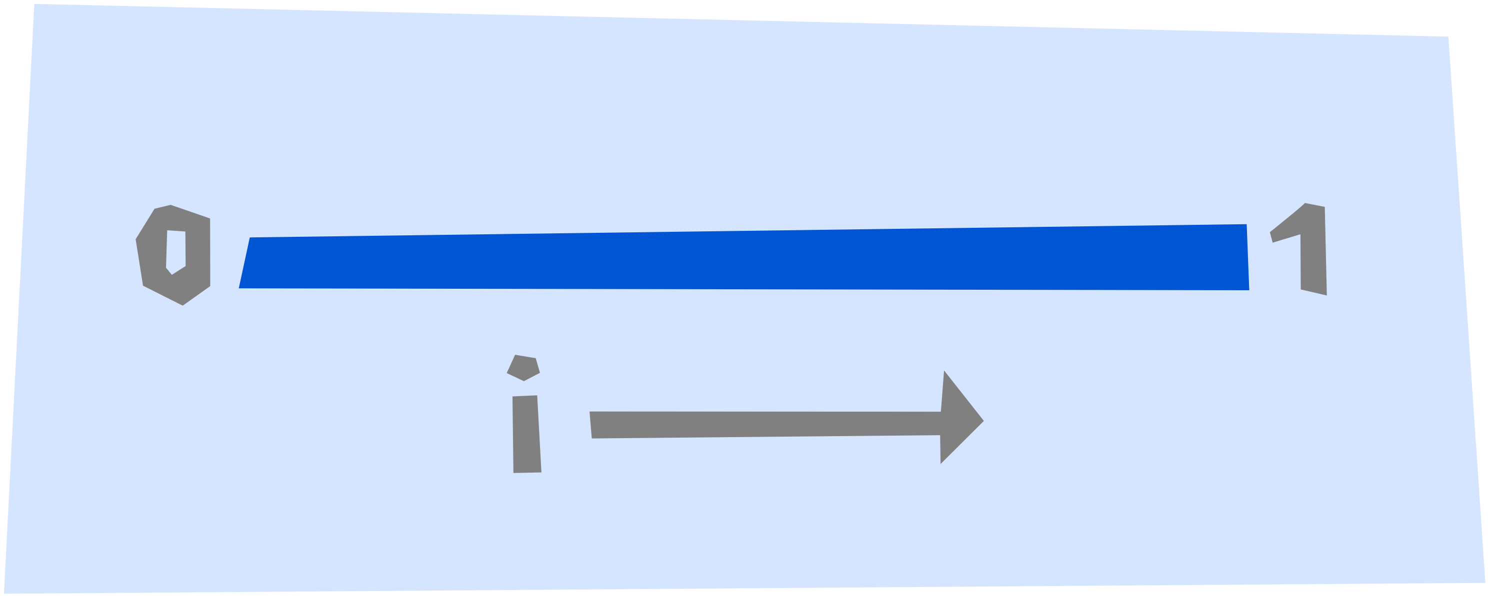
a different composition
based on non-nullable ones

with a different set of equations to avoid regularity

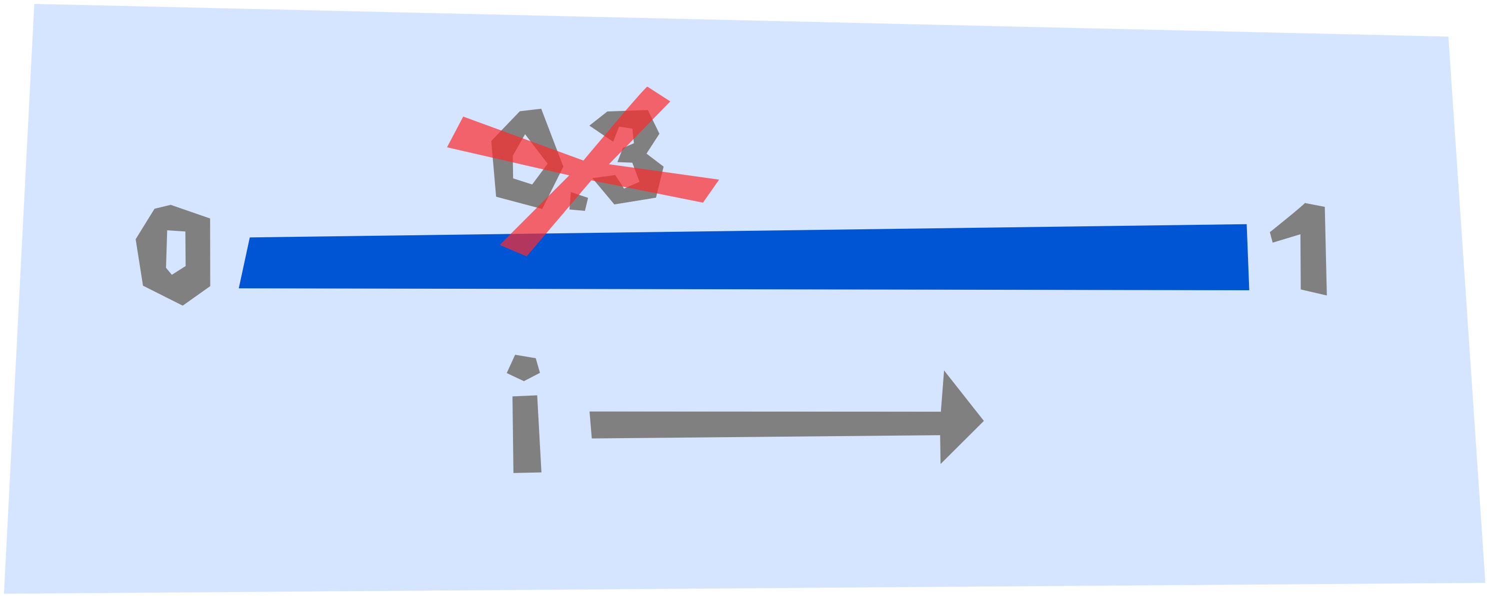
Plan C

a different composition
based on non-nullable ones
with a different set of equations to avoid regularity

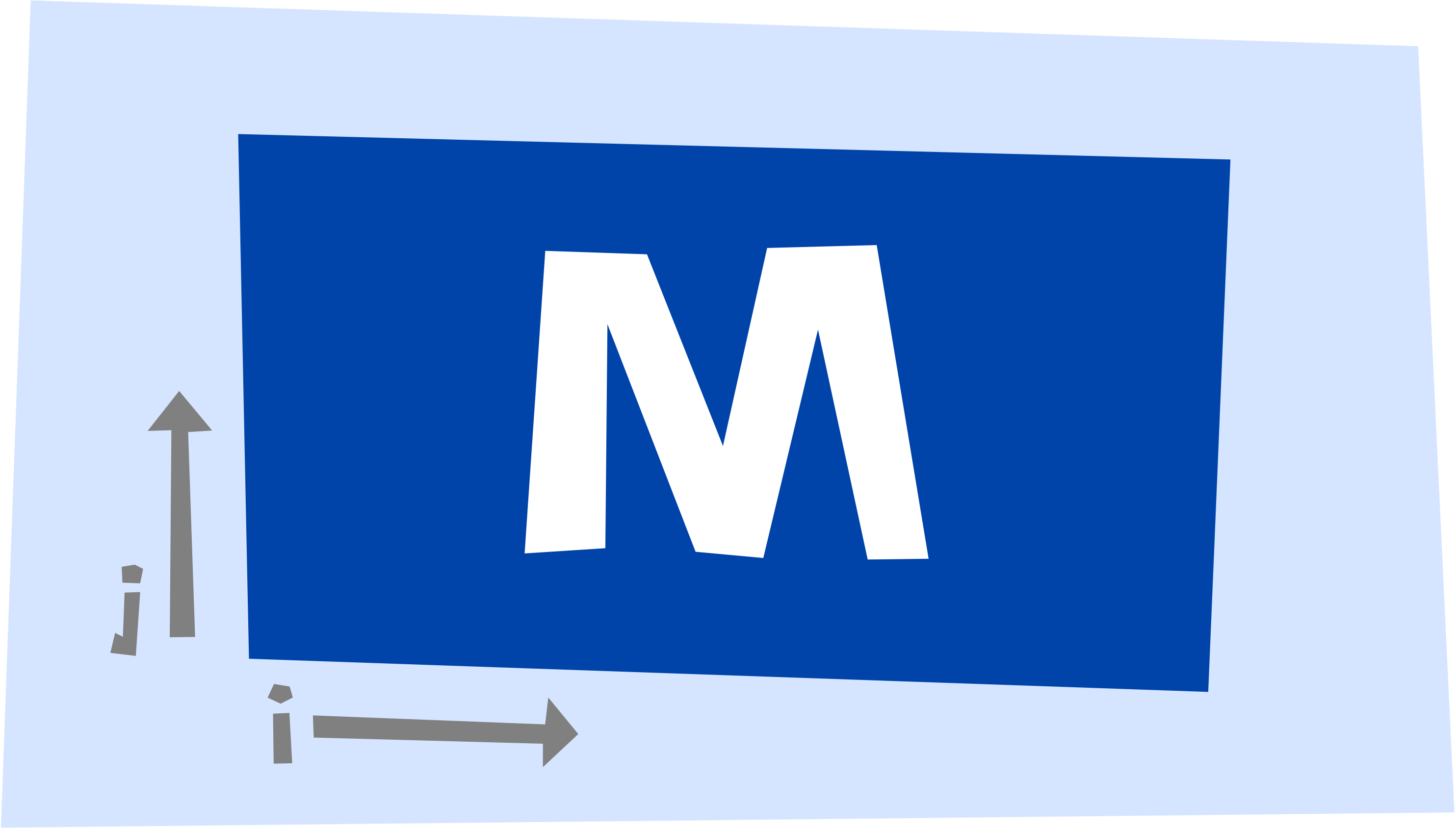
method 1: decision tree
method 2: reflection
no general construction yet



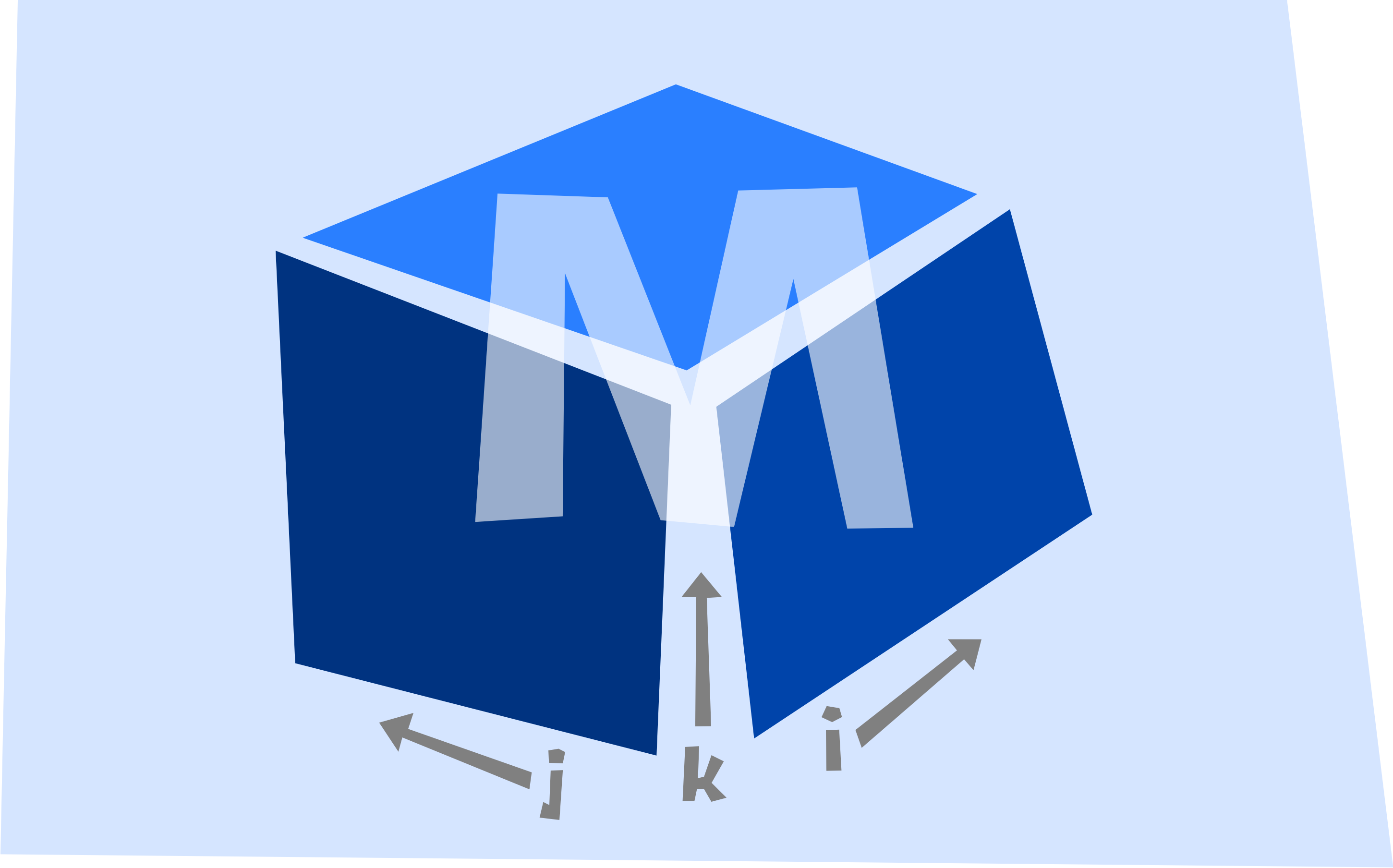
$$i : I \vdash M : A$$



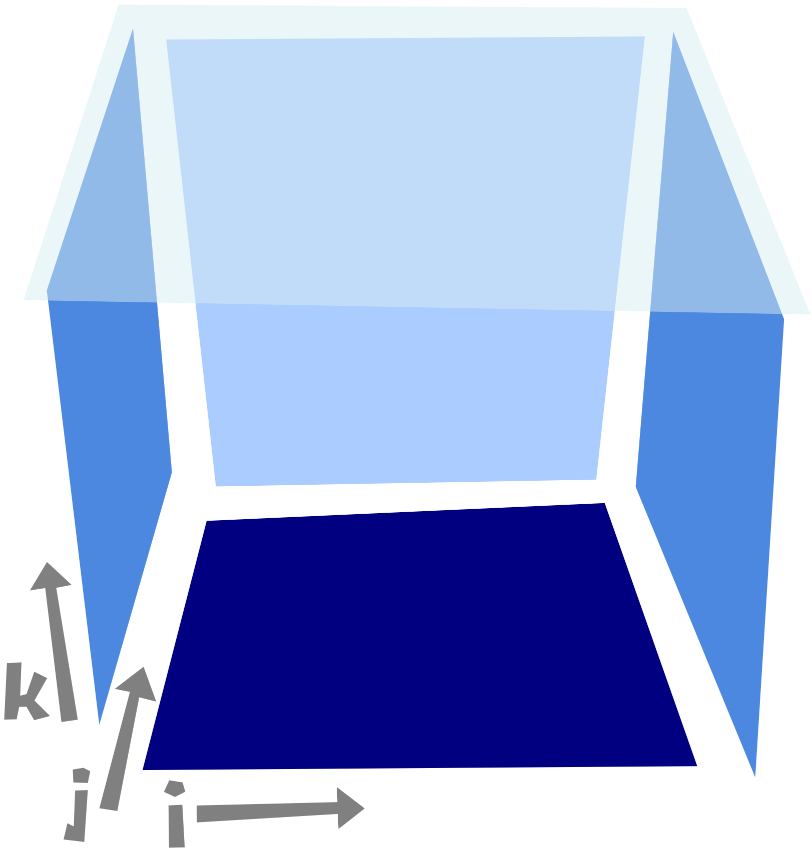
$$i : \mathbb{I} \vdash M : A$$

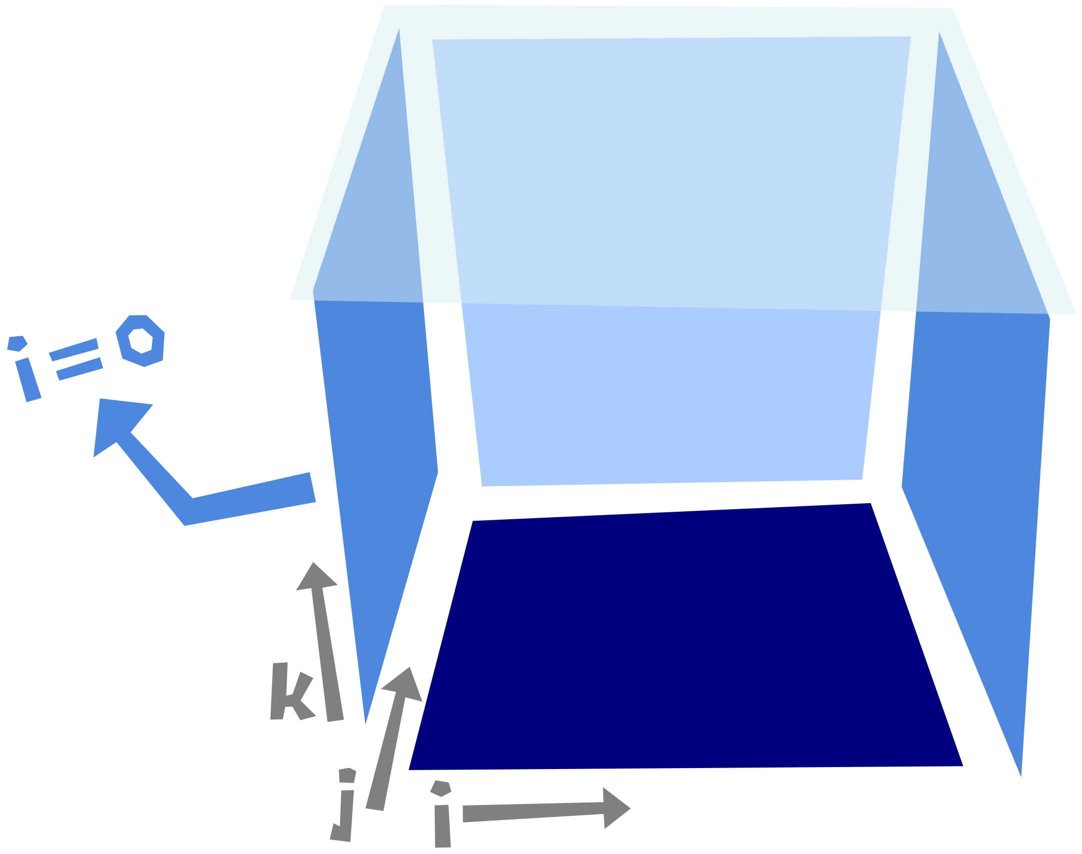


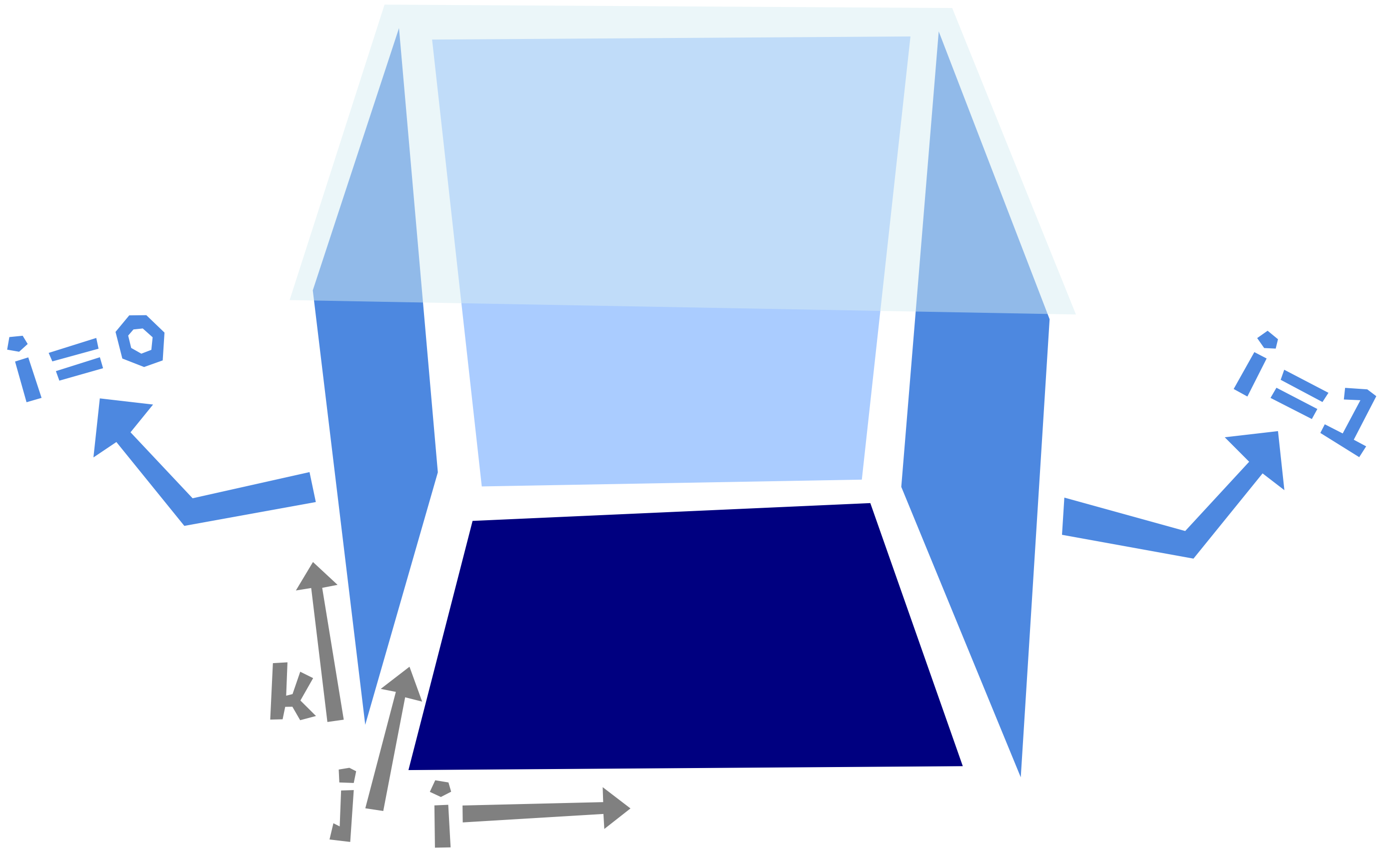
$$i{:}I,\ j{:}I \vdash M{:}A$$

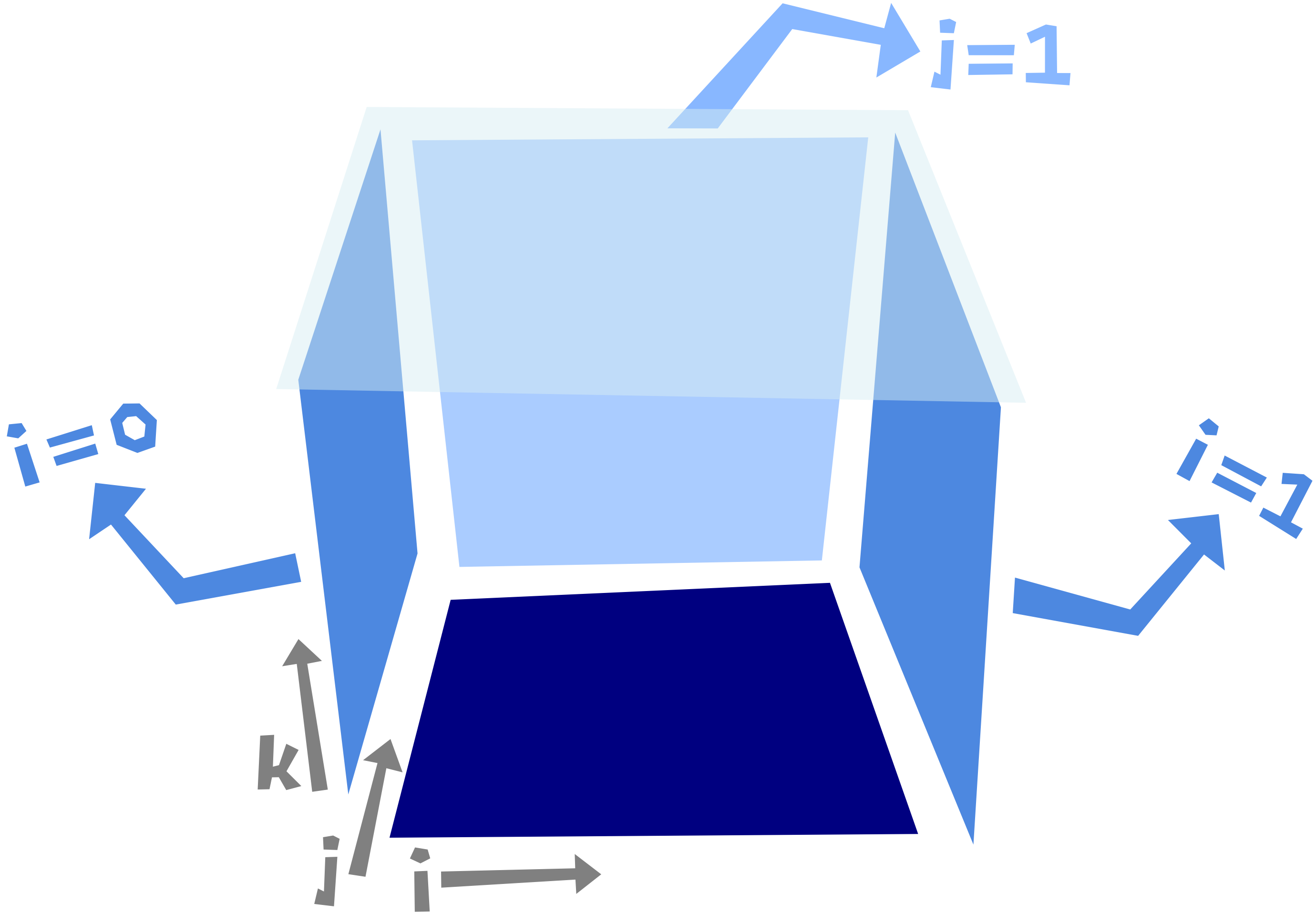


$i:I,\ j:I,\ k:I \vdash M:A$

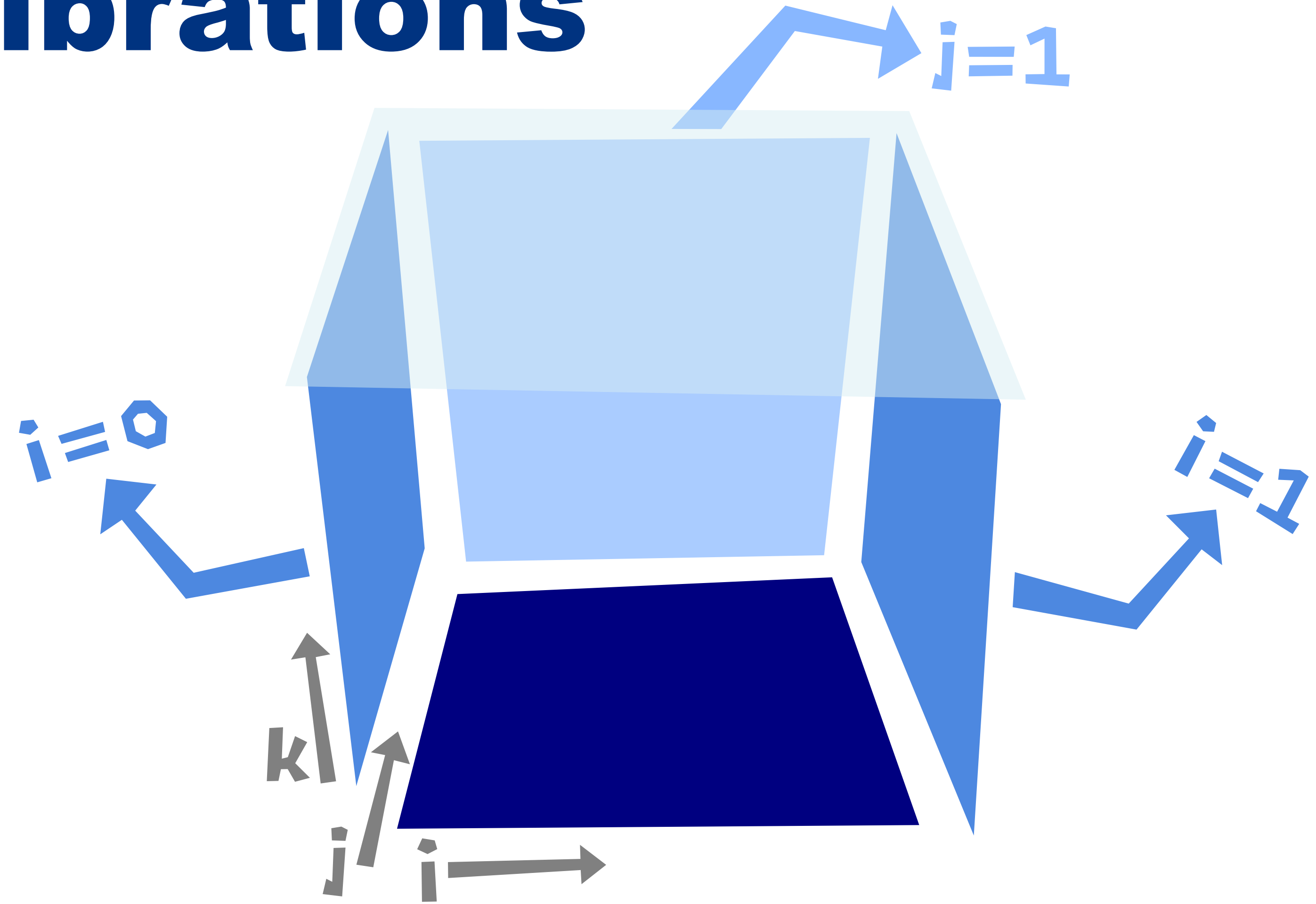

k
j
i


i=0
k
j
i


i=0
i=1
k
j
i

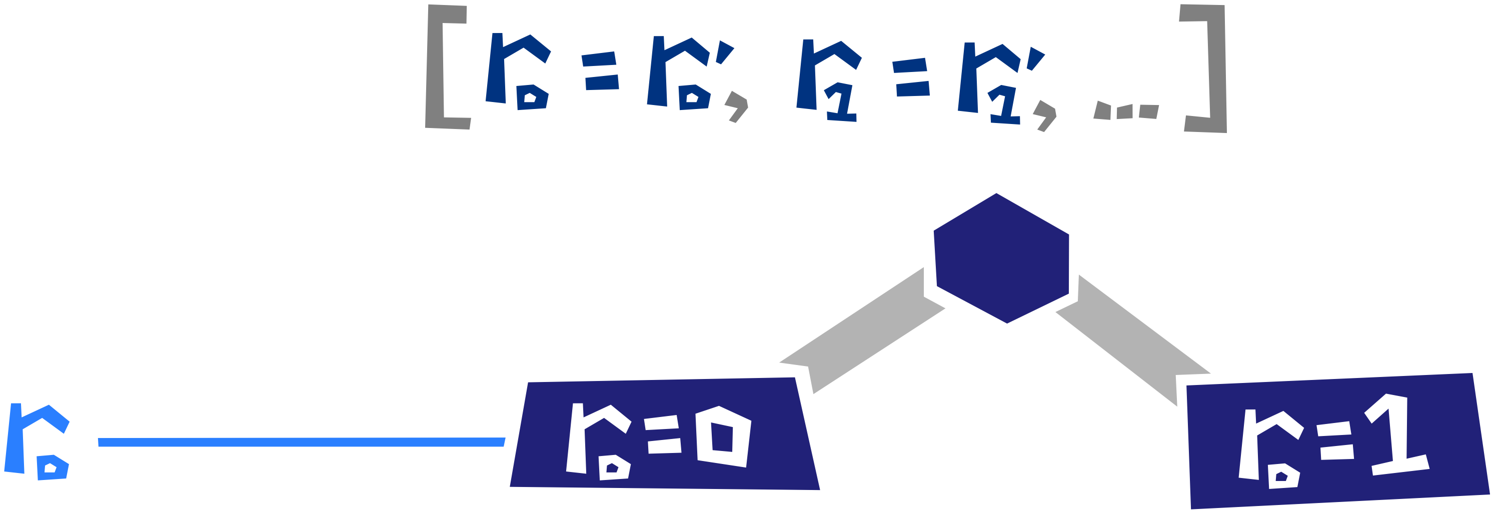

j=1
i=0
i=1
k
j
i

cofibrations

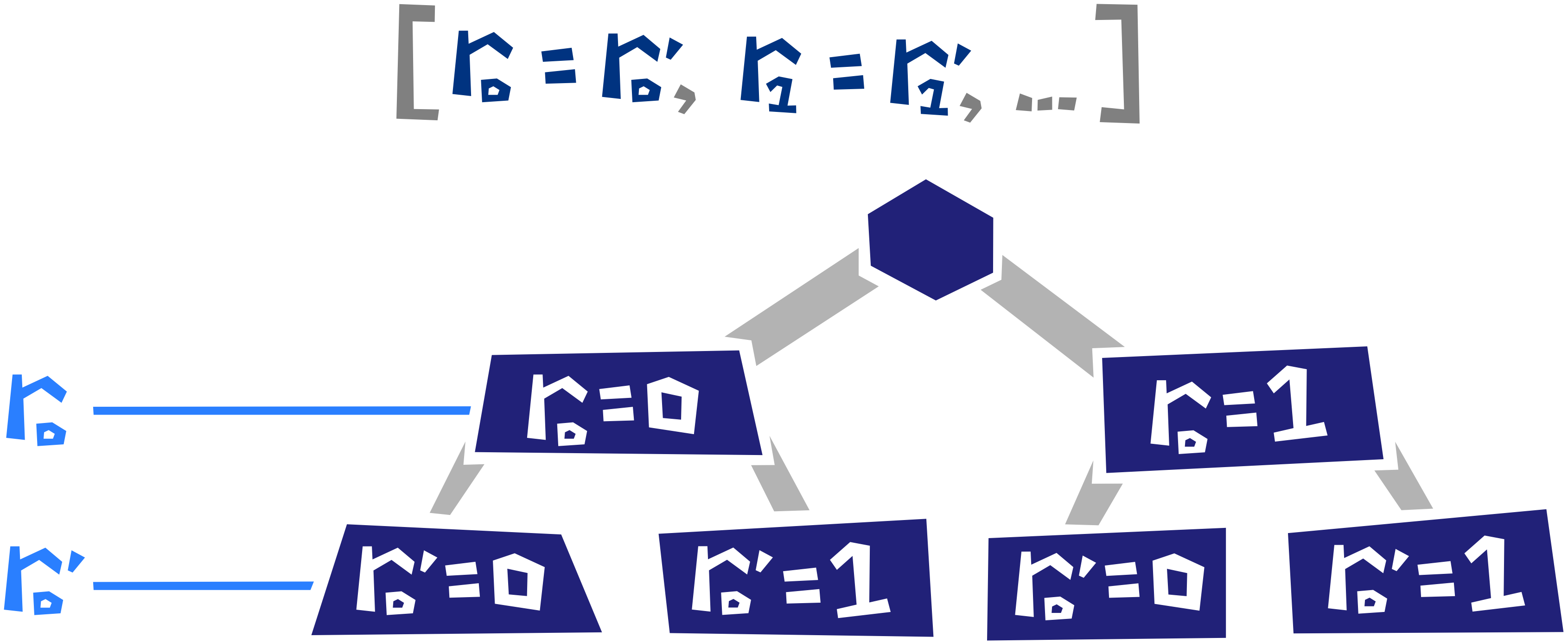


method 1: decision tree

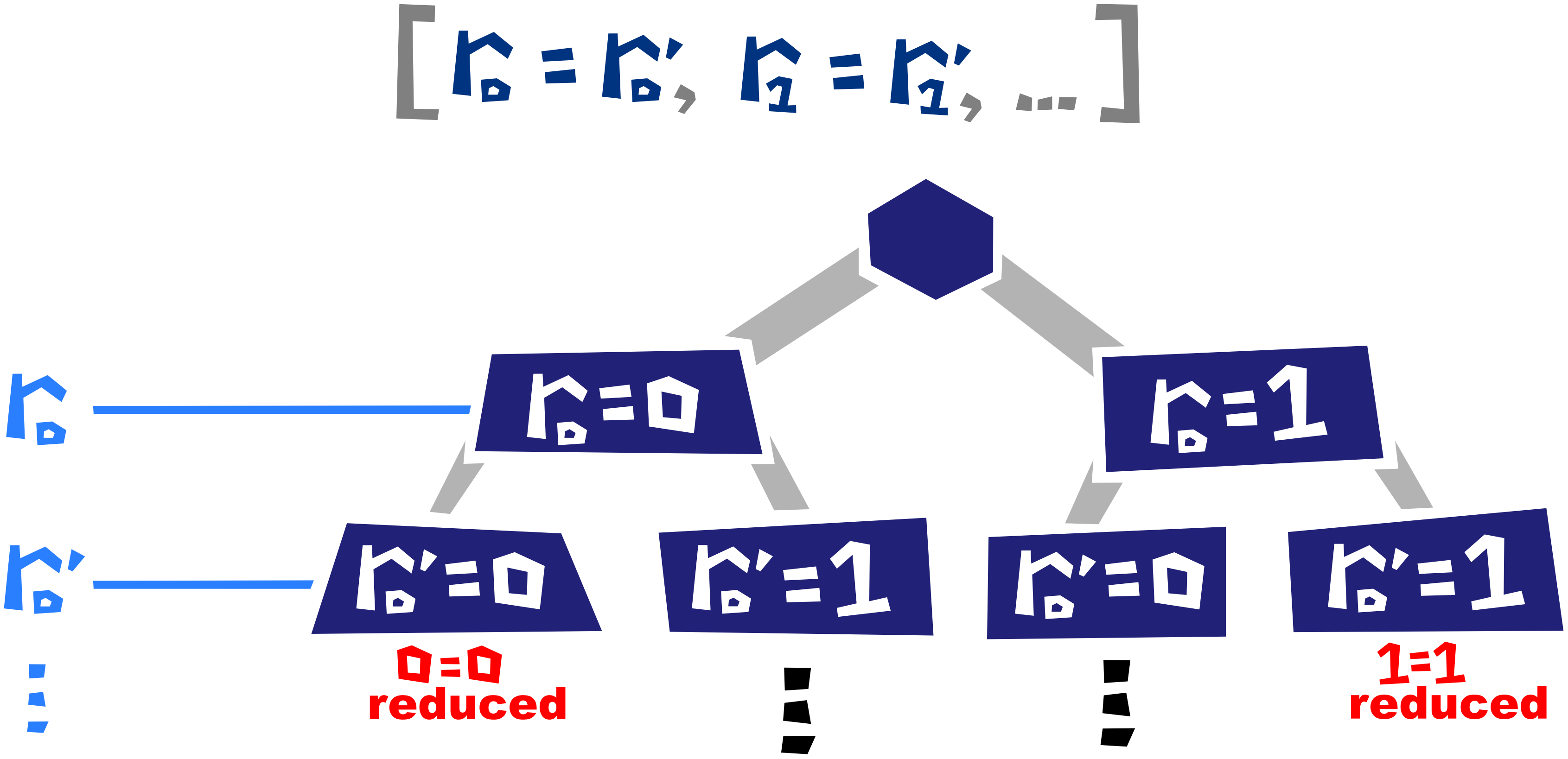
method 1: decision tree



method 1: decision tree



method 1: decision tree

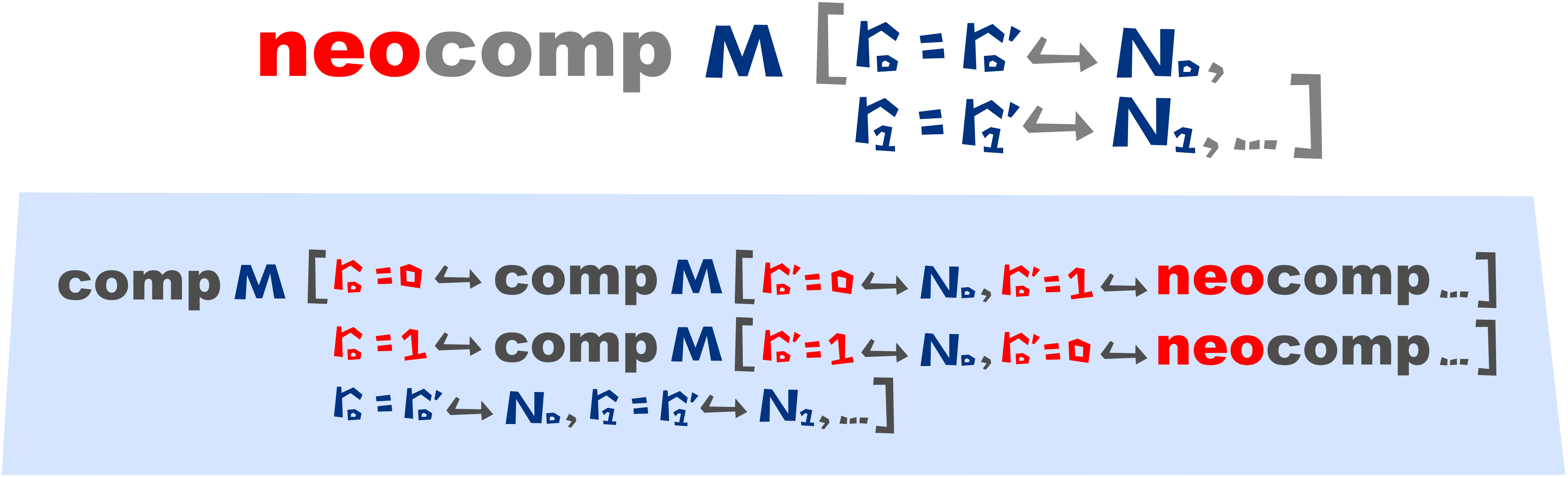


method 1: decision tree

neocomp $M \left[\hat{k}_0 = \hat{k}_0' \hookrightarrow N_0, \; \hat{k}_1 = \hat{k}_1' \hookrightarrow N_1, \ldots \right]$

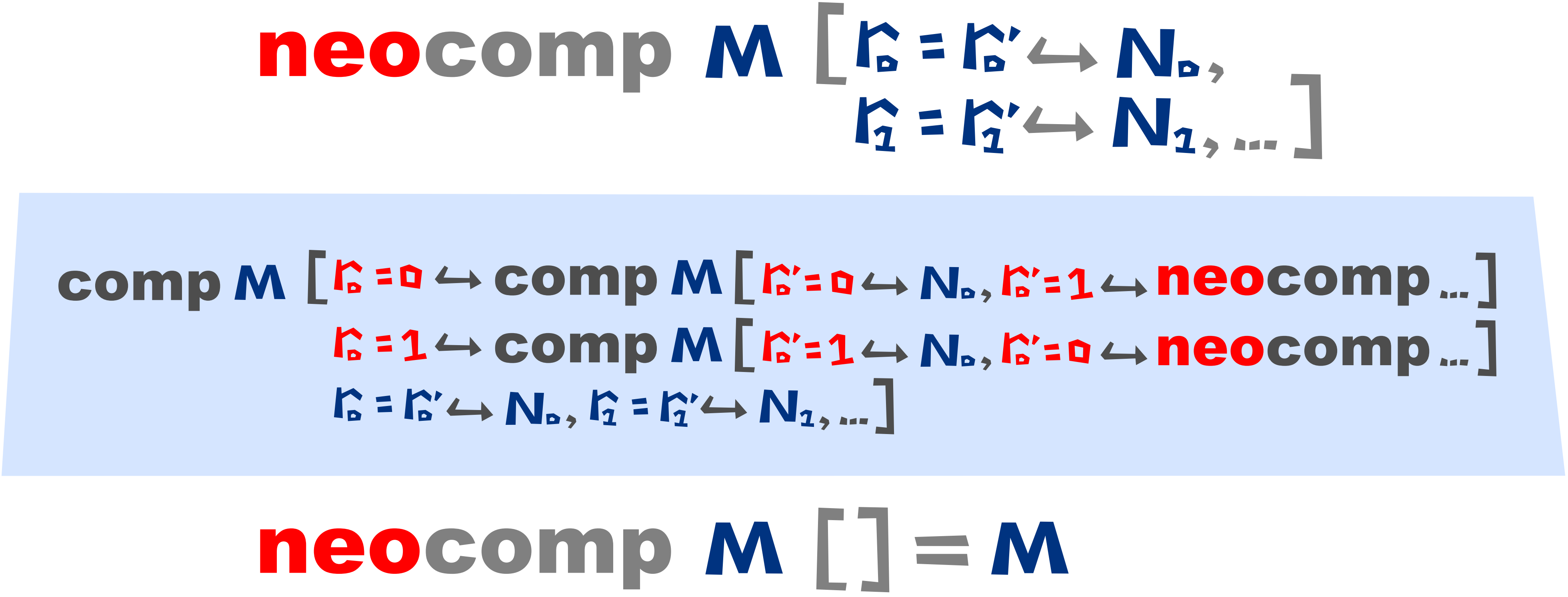
See [AFH] and/or Carlo's thesis

method 1: decision tree



See [AFH] and/or Carlo's thesis

method 1: decision tree



See [AFH] and/or Carlo's thesis

method 1: decision tree

neocomp M [$\hat{n}_0 = \hat{n}_0' \hookrightarrow N_0$,
$\hat{n}_1 = \hat{n}_1' \hookrightarrow N_1$, ...]

limitation: the way/order to check dimension expressions needs to respect all equalities (e.g., subst.)

method 1: decision tree

variants of [AFH]-style composition

D removal of duplicate walls
I removal of inconsistent walls
P permutation of walls
S symmetry of wall constraints
σ symmetry for non-diagonals only

method 1: decision tree

variants of [AFH]-style composition

- **D** removal of **d**uplicate walls
- **I** removal of **i**nconsistent walls
- **P** **p**ermutation of walls
- **S** **s**ymmetry of wall constraints
  - **σ** **s**ymmetry for non-diagonals only

unsolved cases: -P+S

(no permutation, but with symmetry)

method 1: decision tree

[AFH]-style + conjunctions

method 1: decision tree

[AFH]-style + conjunctions

$$\hat{r}_0 = \hat{r}'_0 \land \hat{r}_1 = \hat{r}'_1$$

trickier with +I

how about

$\hat{r} = 0 \land \hat{r} = 1$

method 1: decision tree

[AFH]-style + conjunctions

$\hat{r}_0 = \hat{r}_0' \wedge \hat{r}_1 = \hat{r}_1'$

trickier with +I

how about
$r = 0 \wedge r = 1$

solved case by case

[AFH], research notes, ...

method 2: reflection

[CCHM]-style composition

method 2: reflection

[CCHM]-style composition

make intervals richer so that
f(r)=(r=1)
is surjective

method 2: reflection

neocomp M [r=1→N]

method 2: reflection

neocomp M [r=1 ↪ N]

comp M [r=1 ↪ N
r=0 ↪ M]

method 2: reflection

neocomp M [r=1 ↪ N]

comp M [r=1 ↪ N
r=0 ↪ M]

used in Cubical Agda

Plan C

a different composition based on non-nullable ones

with a different set of equations to avoid regularity

but, is it worth it?

none works for unknown cofibrations

none works for unknown cofibrations

```
def mycom/fun
  (A : 𝕀 → type) (B : 𝕀 → type)
  (com/A : (r : 𝕀) (φ : 𝔽) (p : (i : 𝕀) (_ : [i=r ∨ φ]
  (com/B : (r : 𝕀) (φ : 𝔽) (p : (i : 𝕀) (_ : [i=r ∨ φ]
  (r : 𝕀) (φ : 𝔽) (p : (i : 𝕀) (_ : [i=r ∨ φ]) (_ : A
```

we can quantify over cofibrations in cooltt

no known way to kill nullable compositions

general theory?

general theory?

build univalent Kan universes with only these cofibrations

$$\{\varphi \mid \neg\neg[\varphi]\}$$

still very open

further reading

[Angiuli] thesis

Computational Semantics of Cartesian Cubical Type Theory

[VMA]

Cubical Agda: a dependently typed programming language with univalence and higher inductive types