Phase Transition in the 0-1 Knapsack Problem

Hassan Andrabi

03 March 2025

Motivation

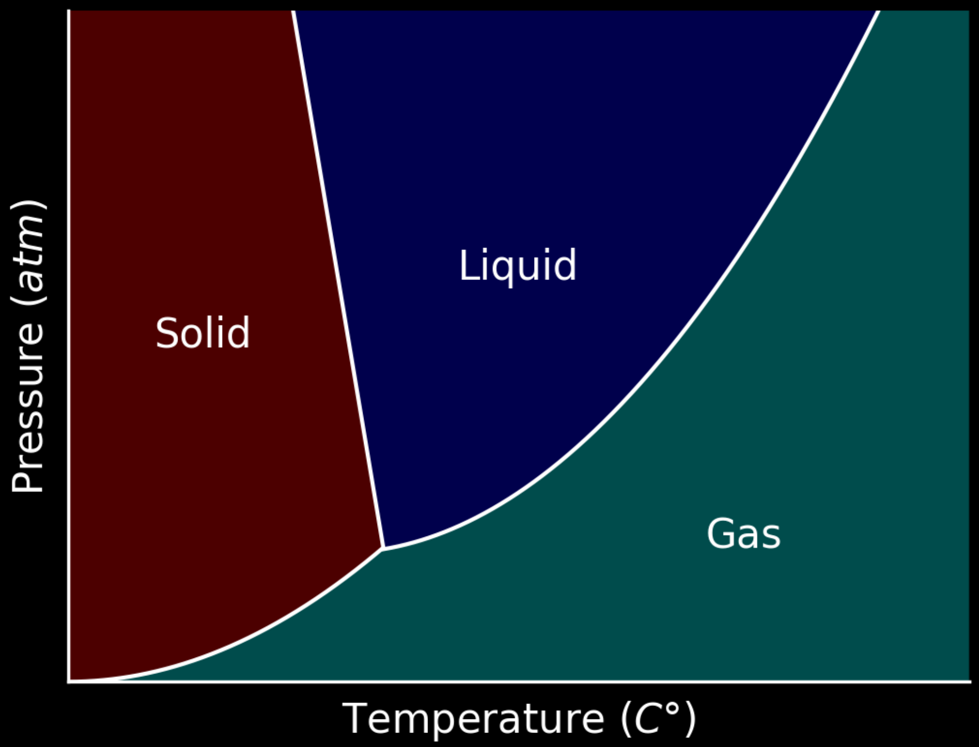
Phase transitions exist in decision problems

What about optimisation problems?

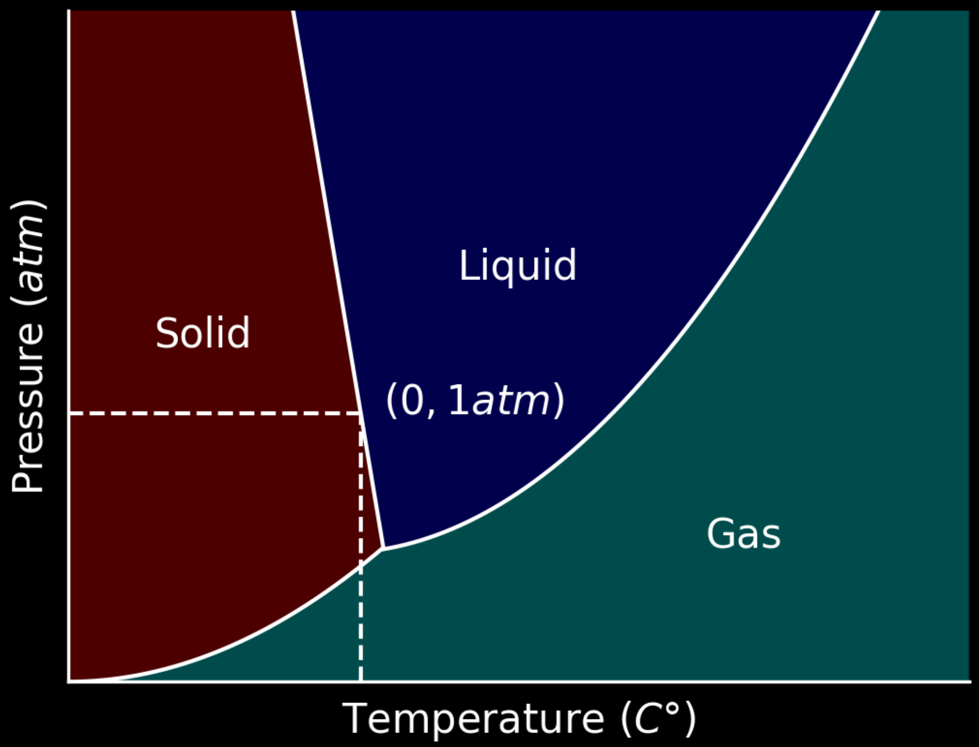
1. Phase transitions
2. Knapsack (decision)
3. Knapsack (optimisation)

1. Phase transitions
2. Knapsack (decision)
3. Knapsack (optimisation)

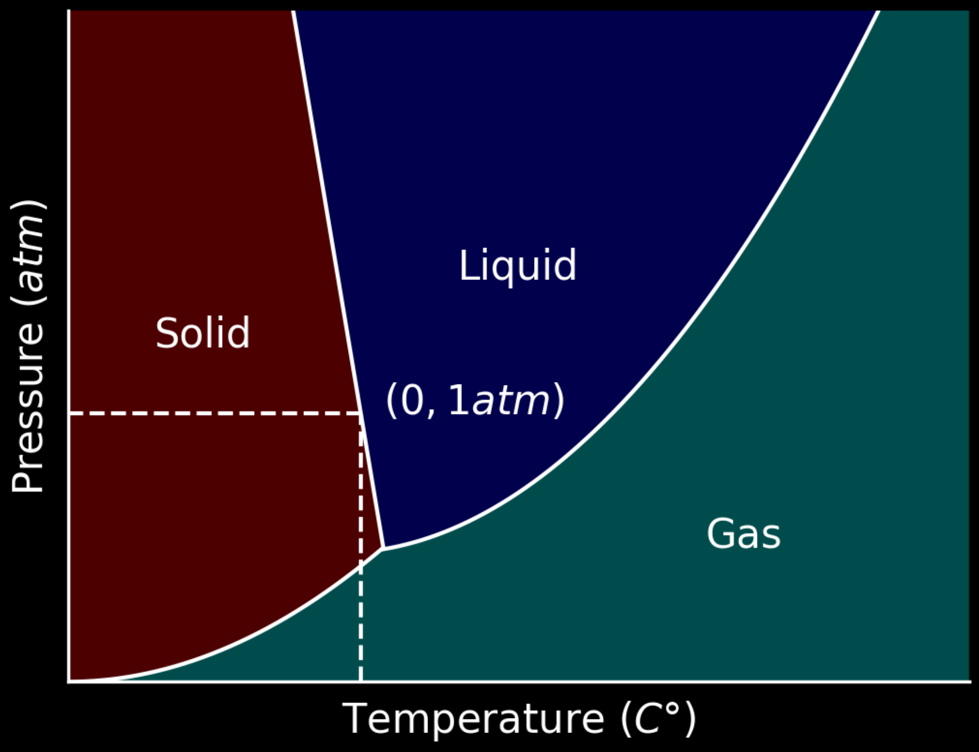
Rapid change in ‘phase’



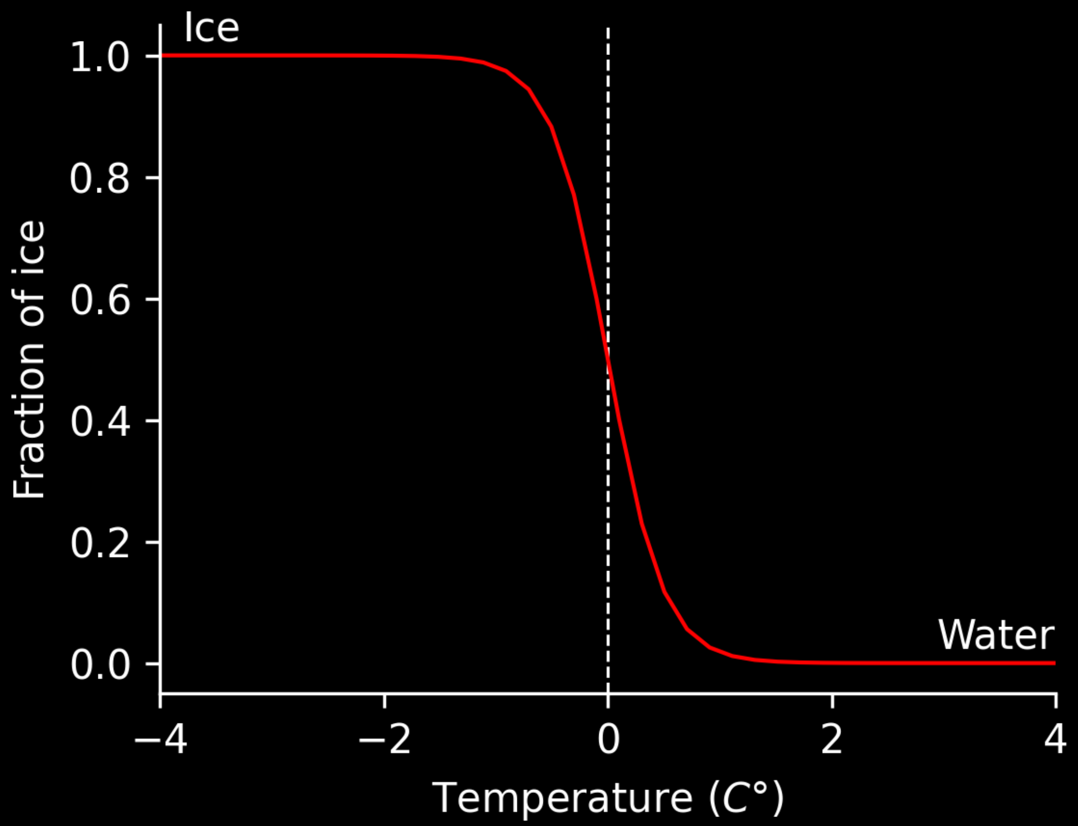
Rapid change in ‘phase’



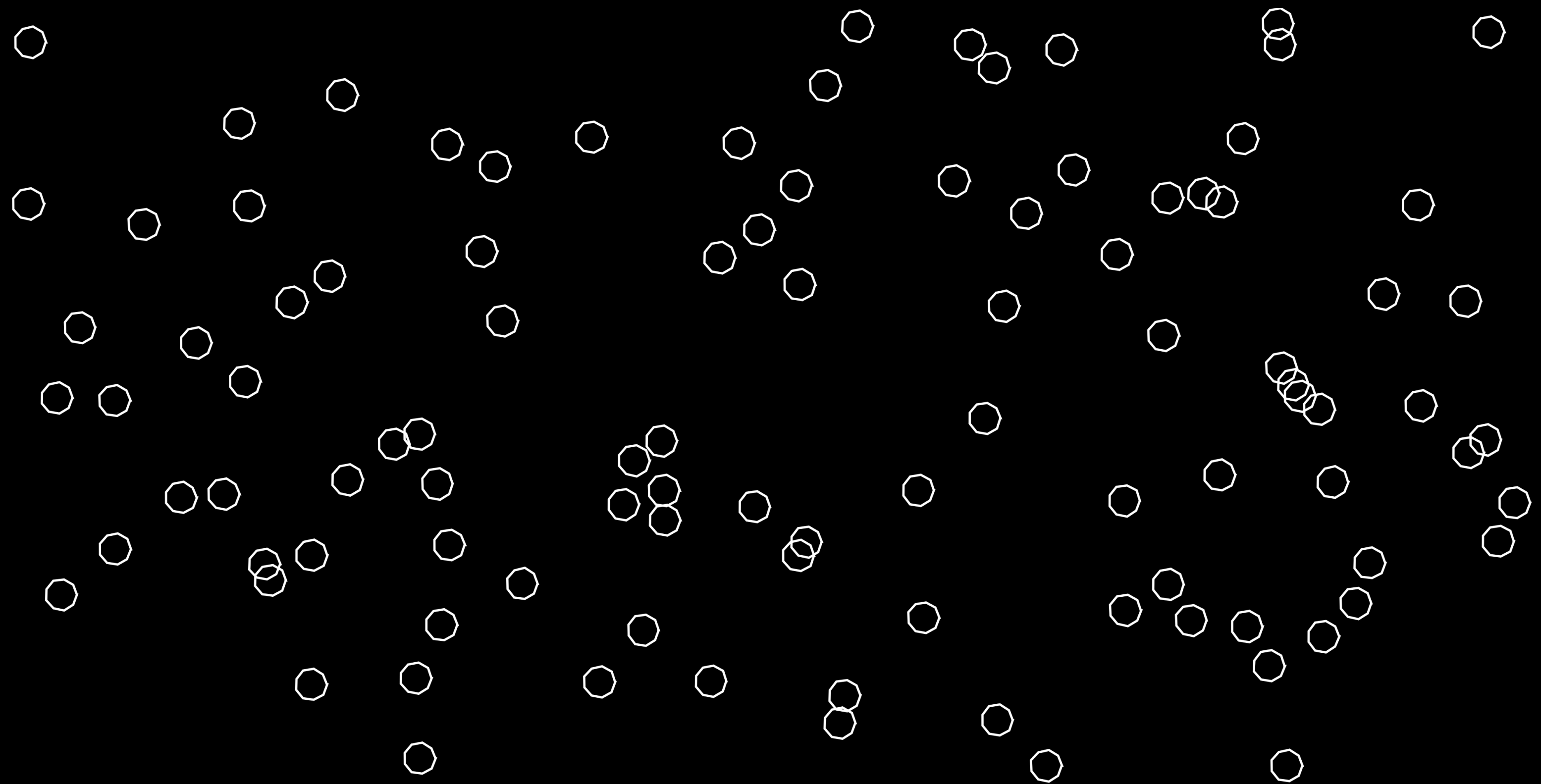
Rapid change in 'phase'



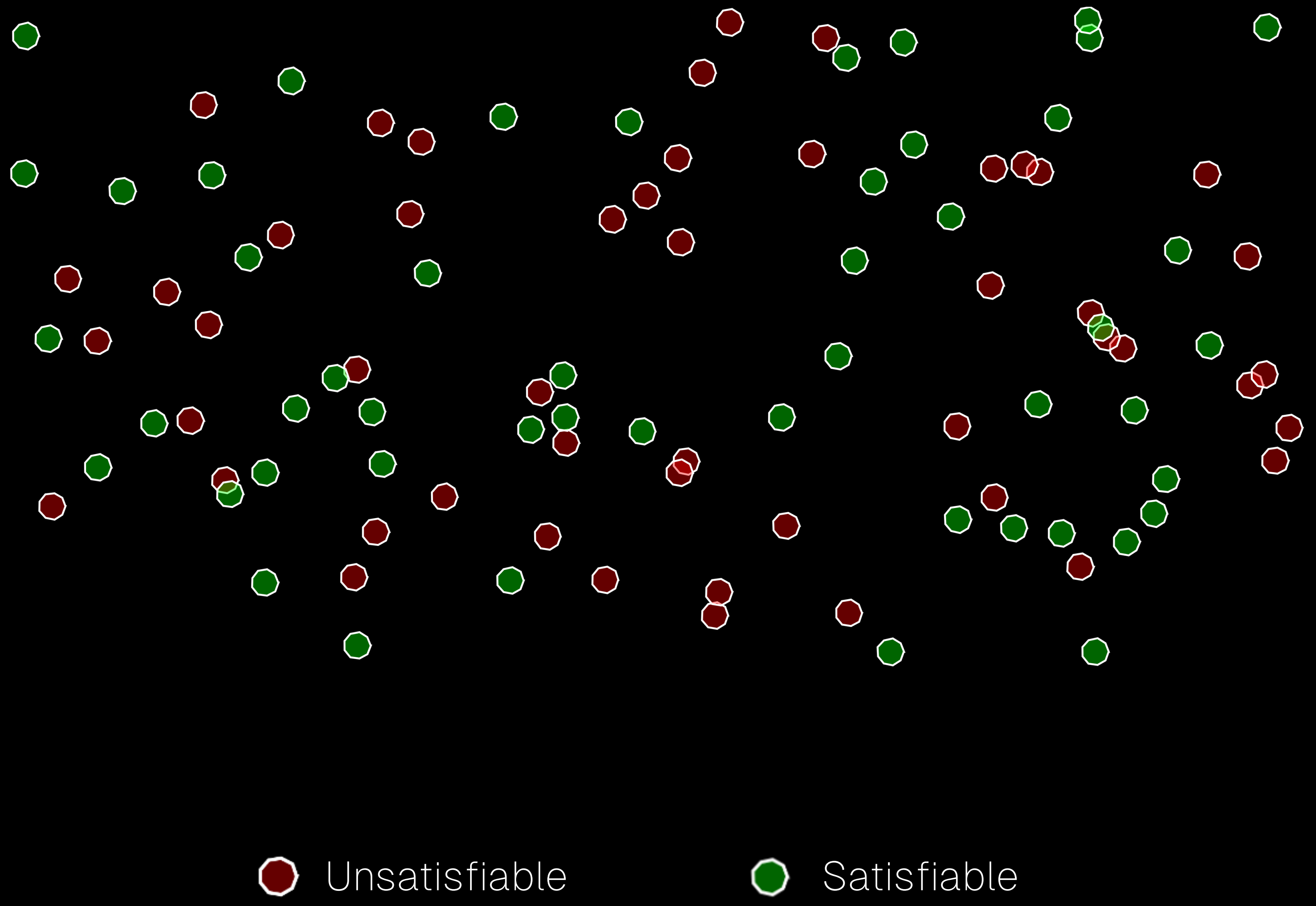
Fixed pressure (1 atm)



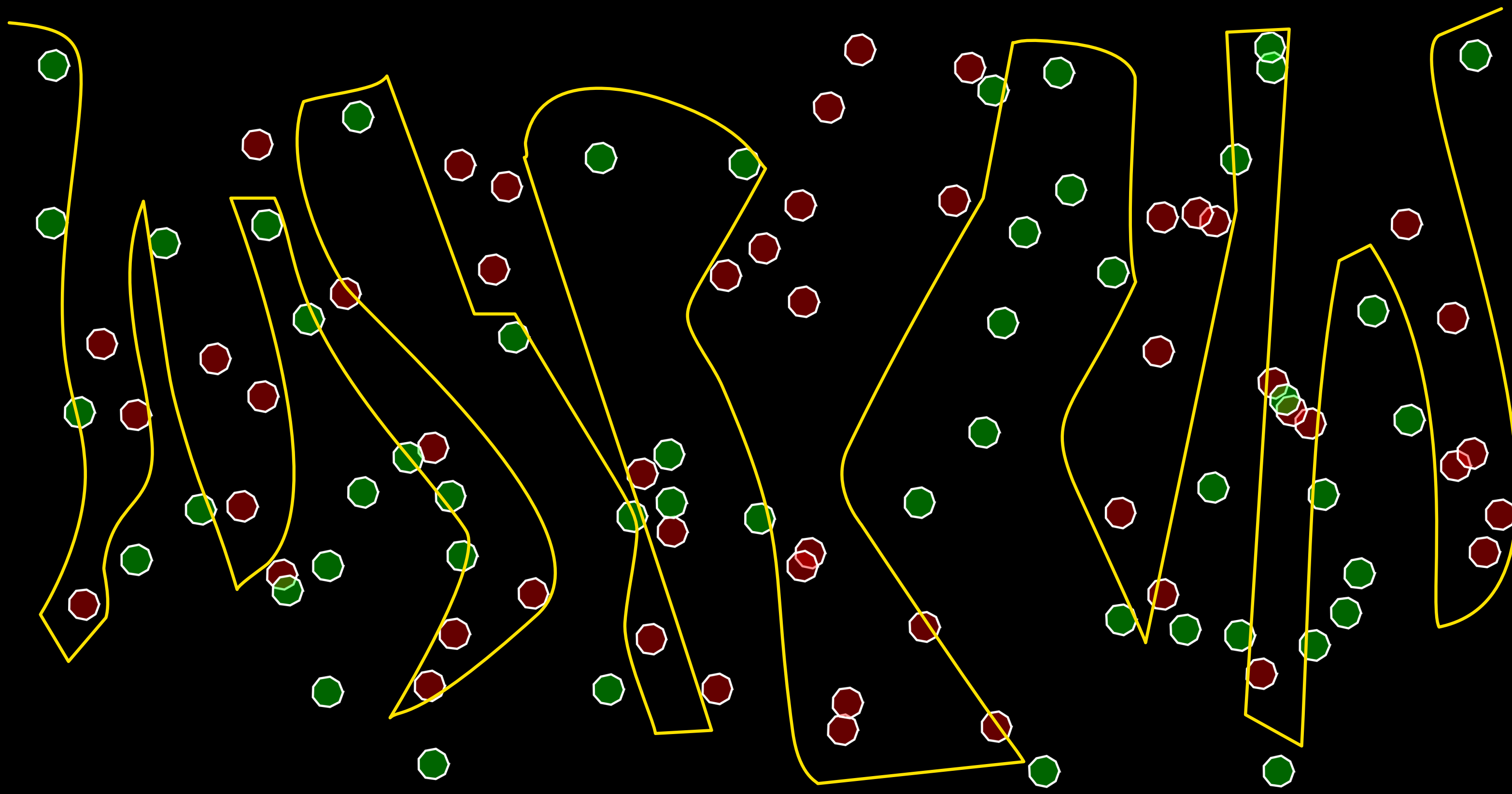
Phases in computational problems

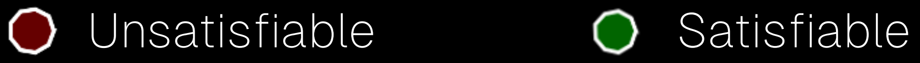


Phases in computational problems

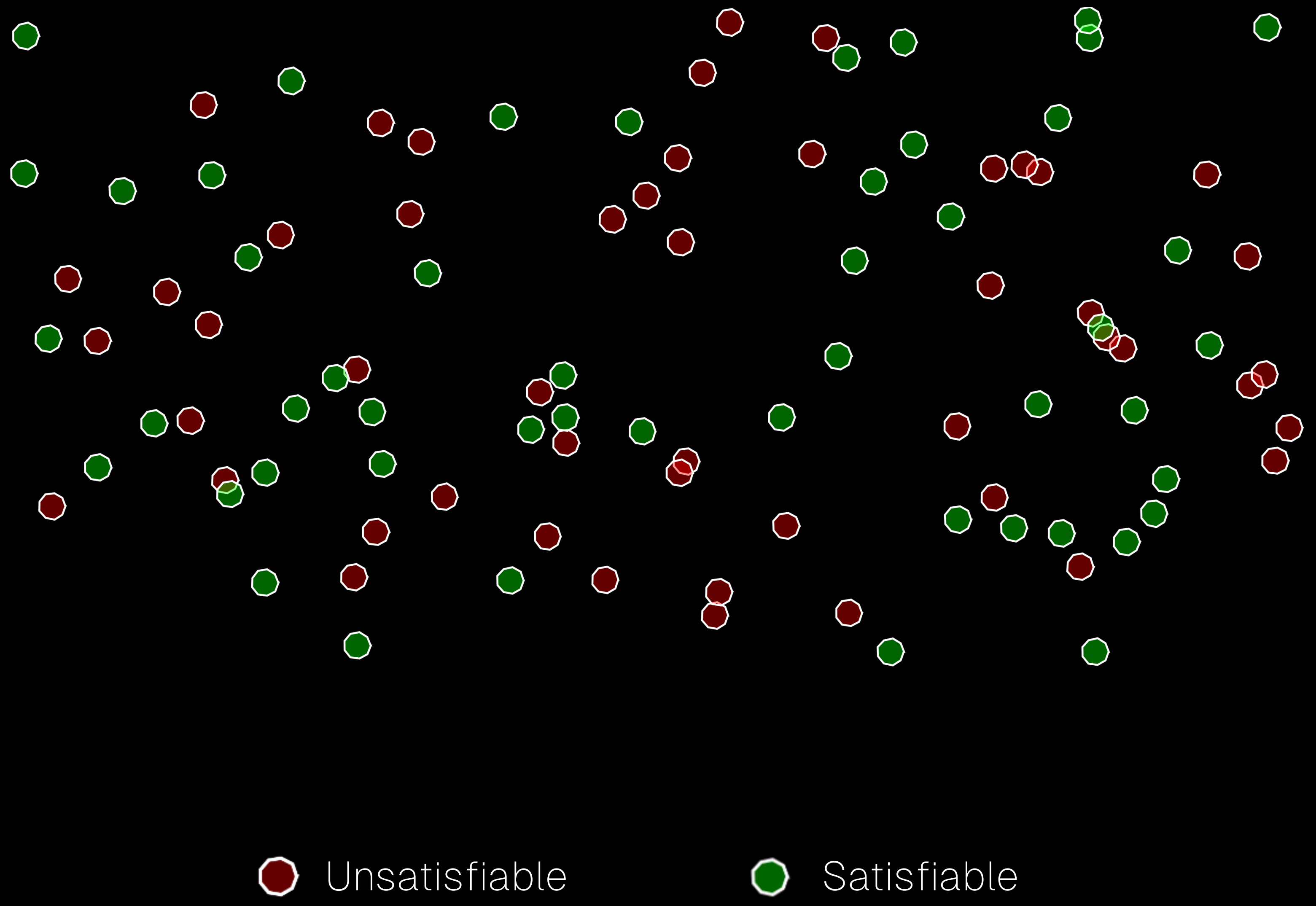


Phases in computational problems

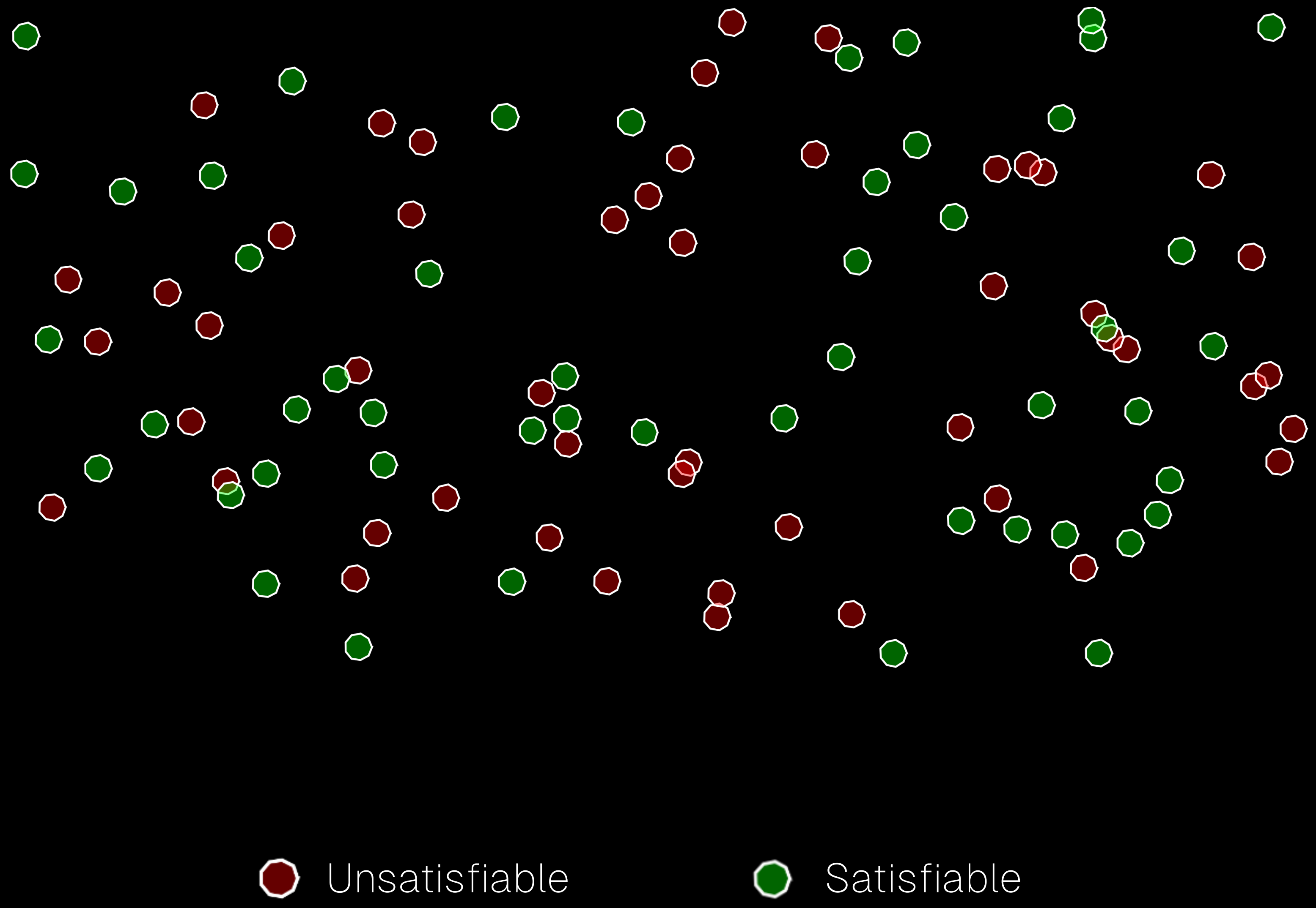




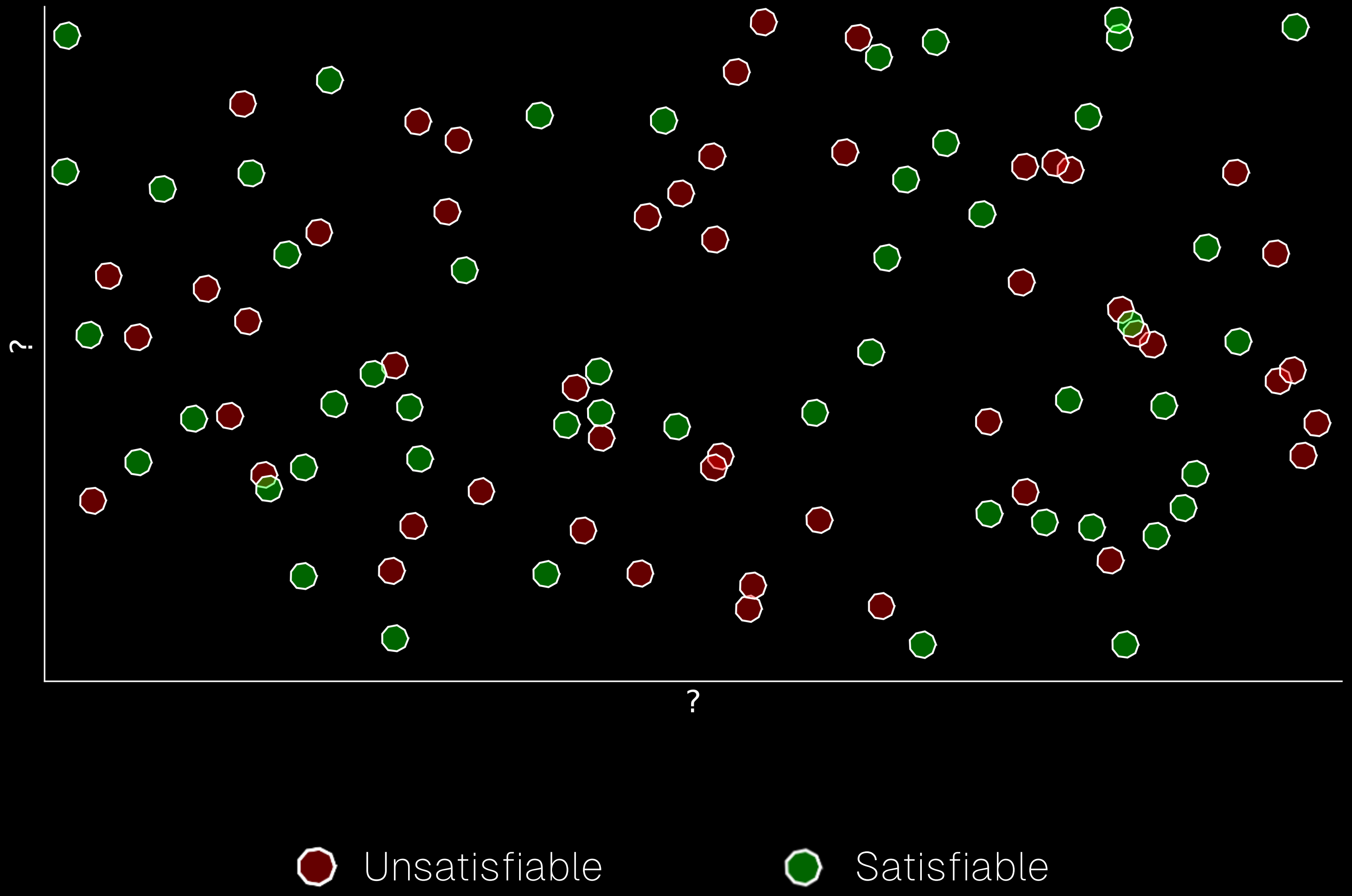
Finding a ‘good’ boundary



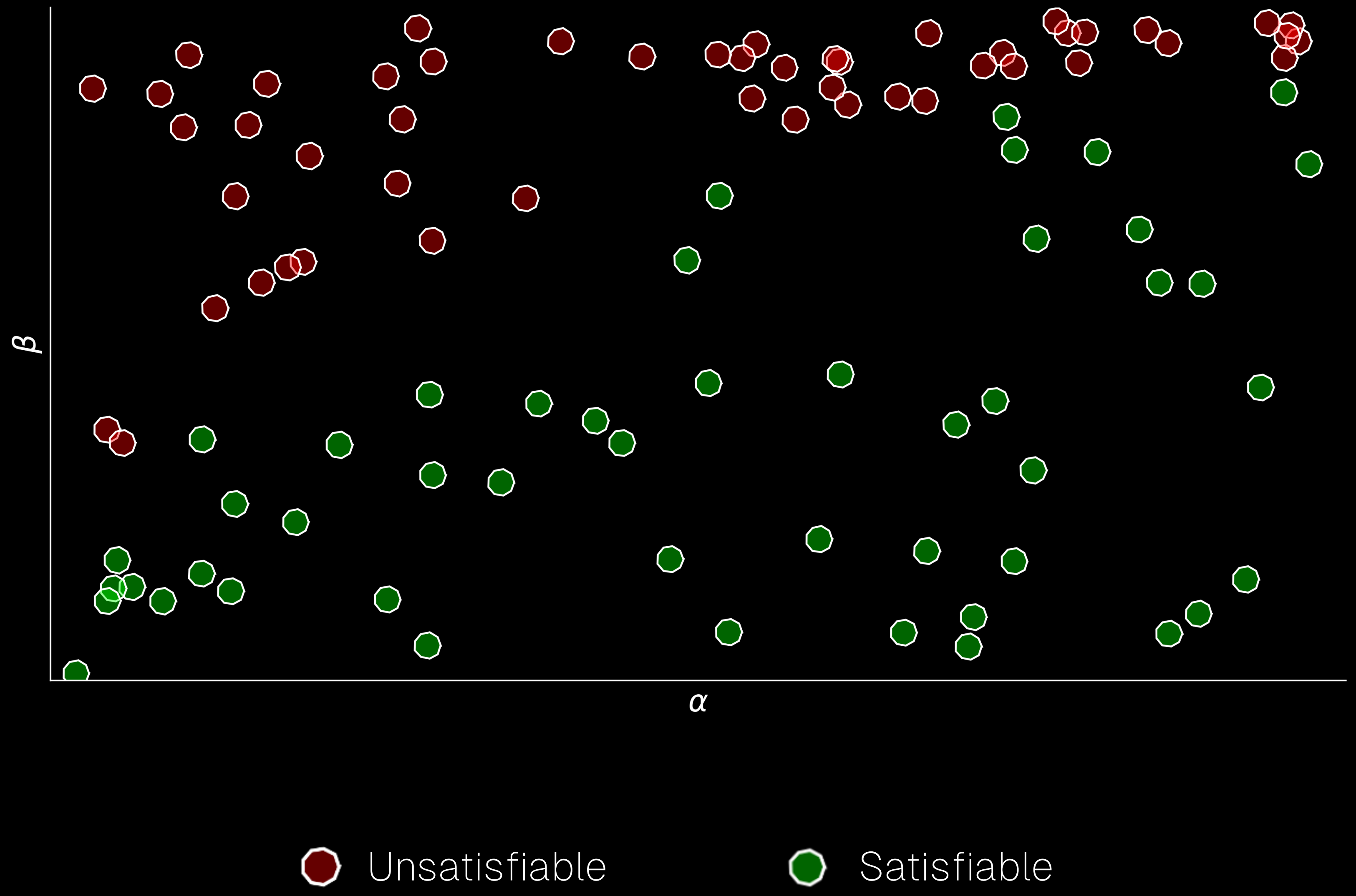
Finding ~~a ‘good’ boundary~~ order parameters



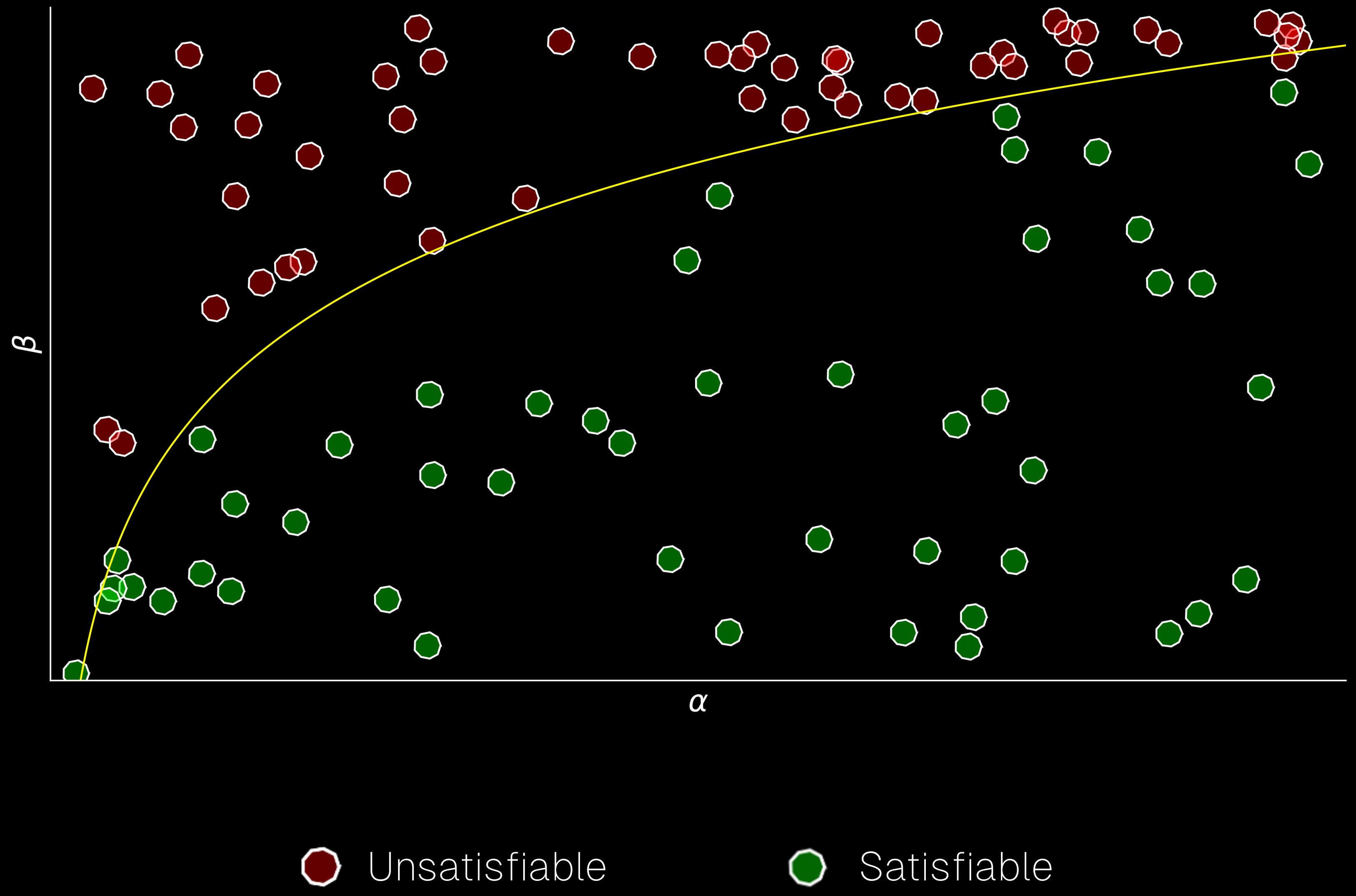
Finding ~~a 'good' boundary~~ order parameters



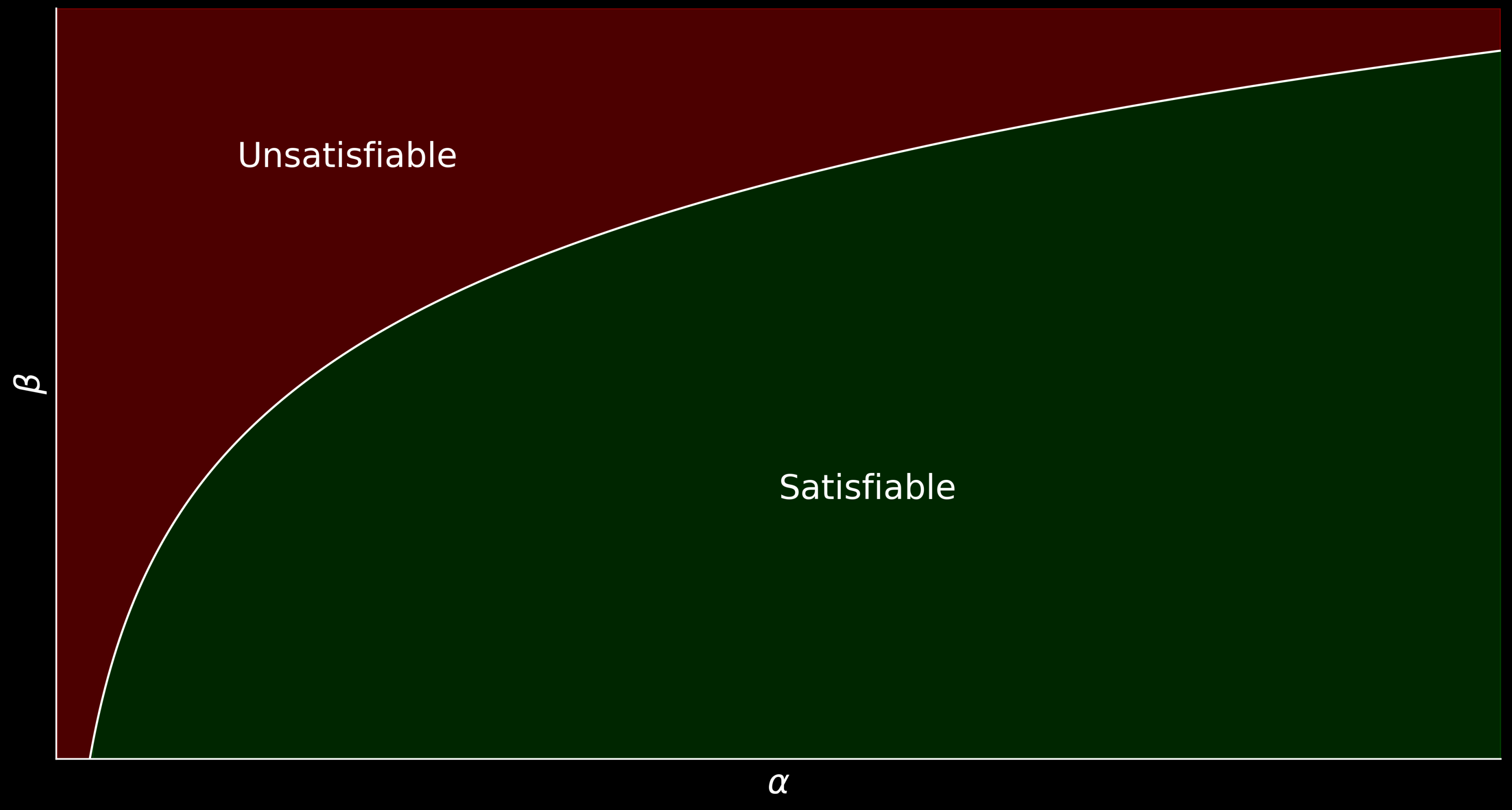
Finding ~~a 'good' boundary~~ order parameters



Finding ~~a 'good' boundary~~ order parameters

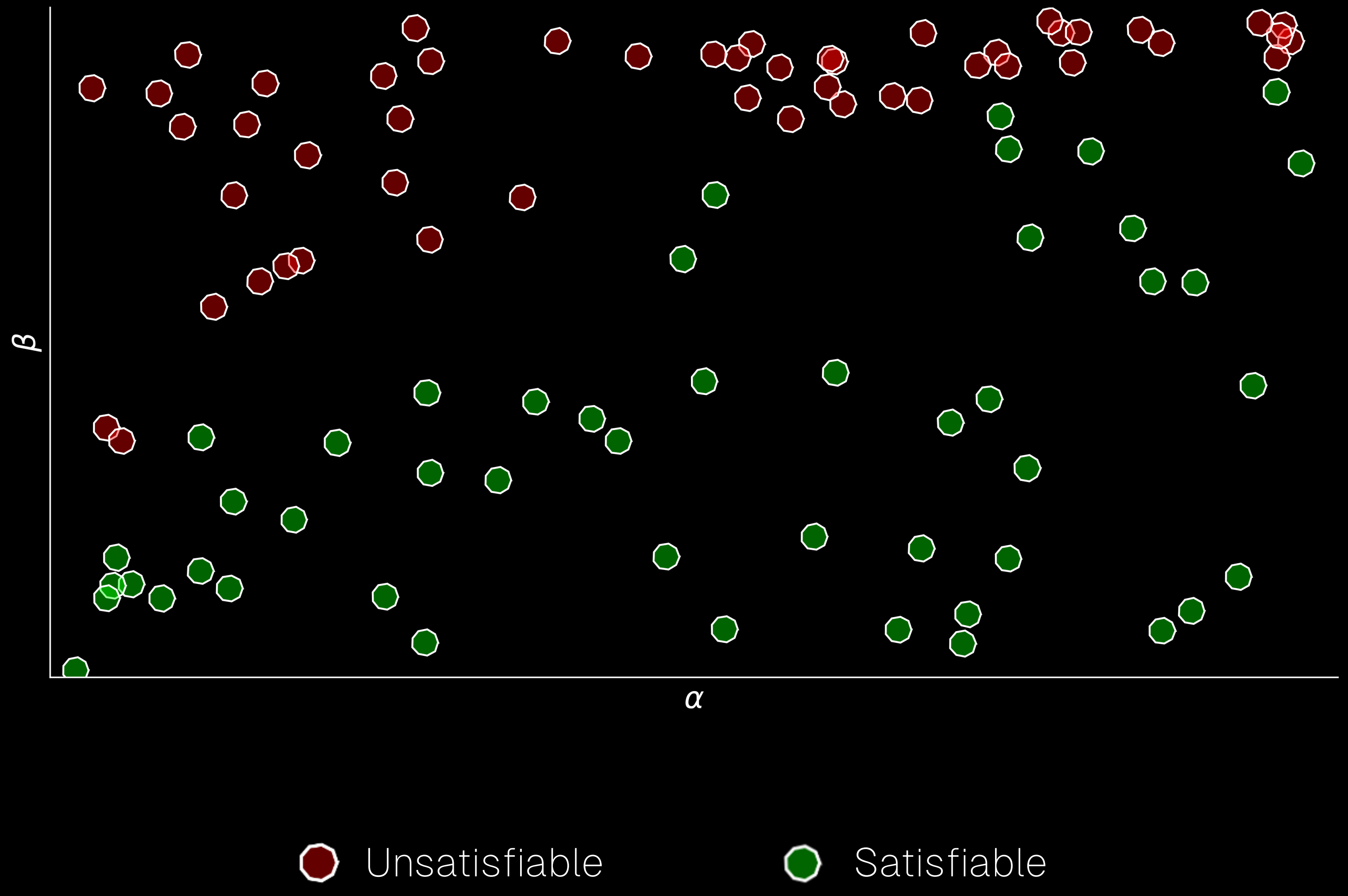


Finding ~~a ‘good’ boundary~~ order parameters

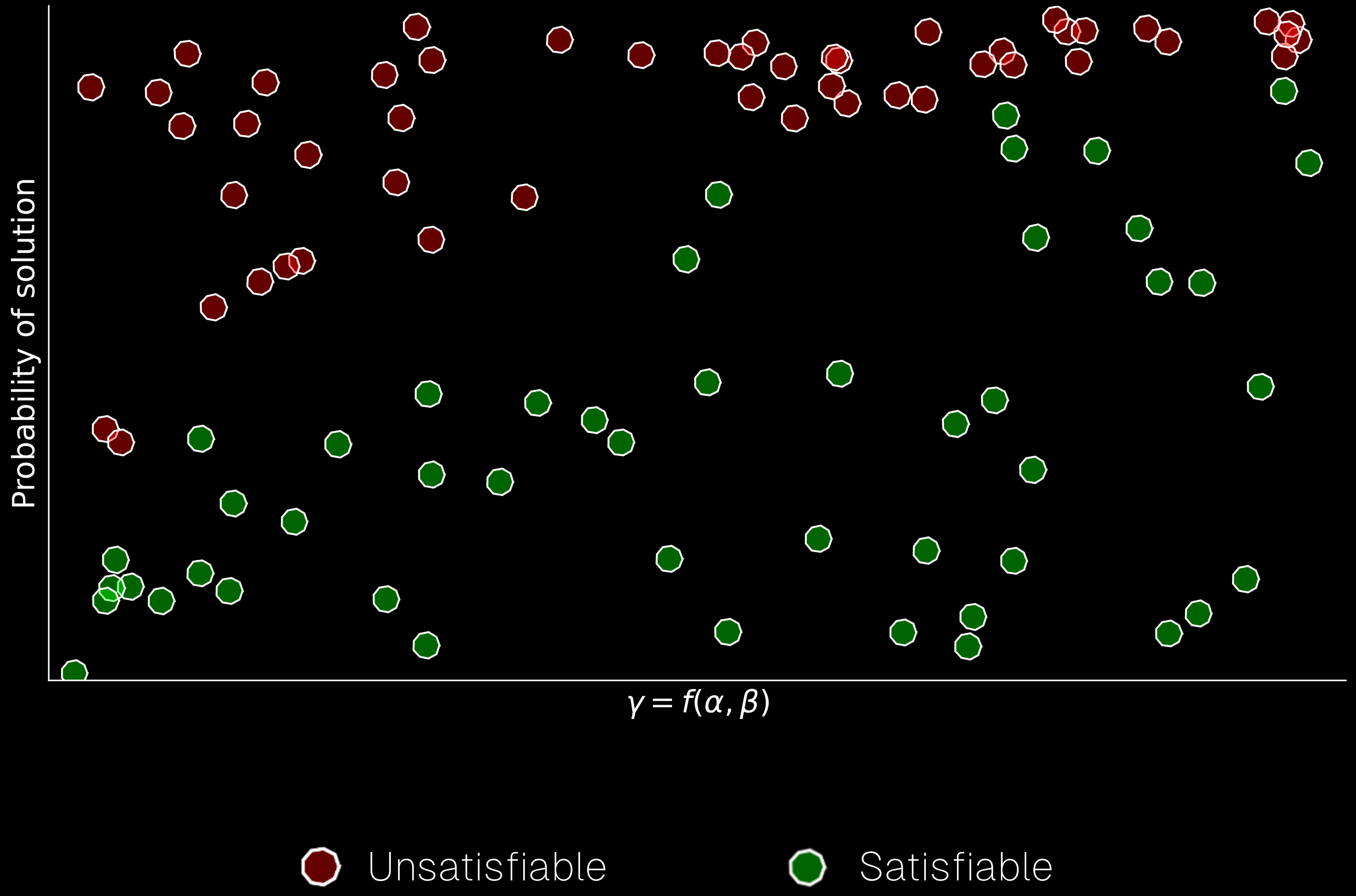


Unsatisfiable Satisfiable

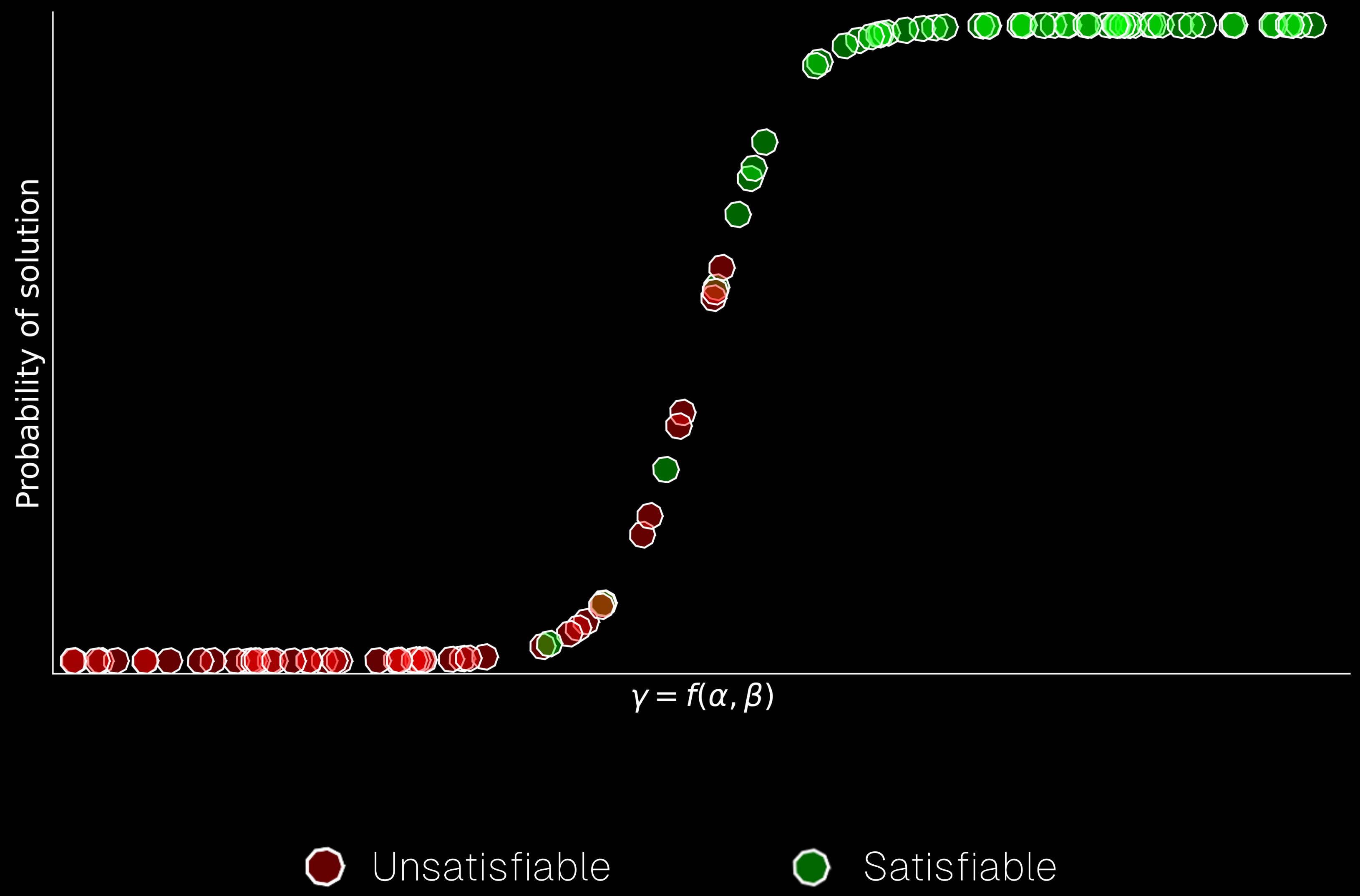
Reducing dimensions



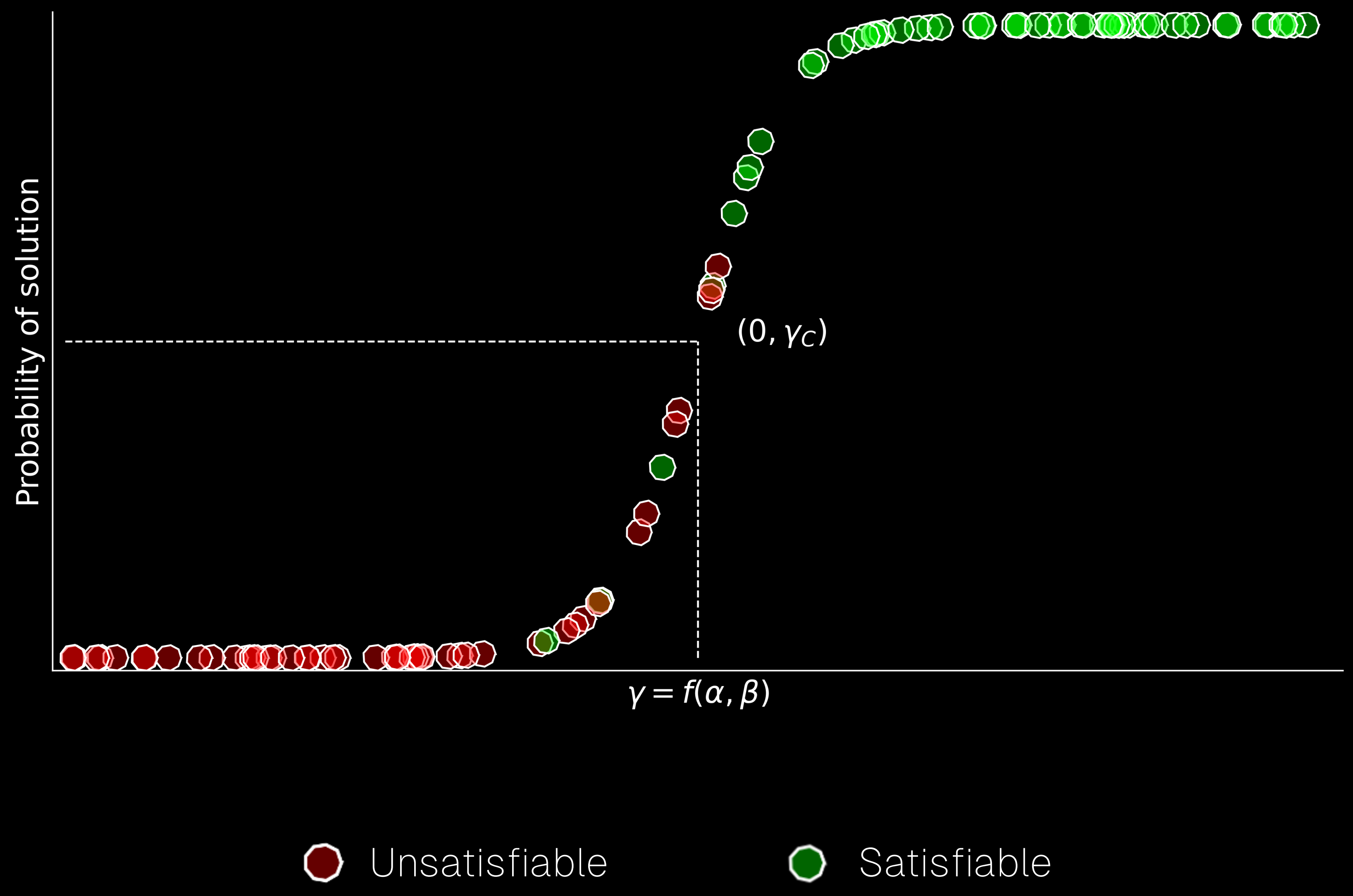
Reducing dimensions



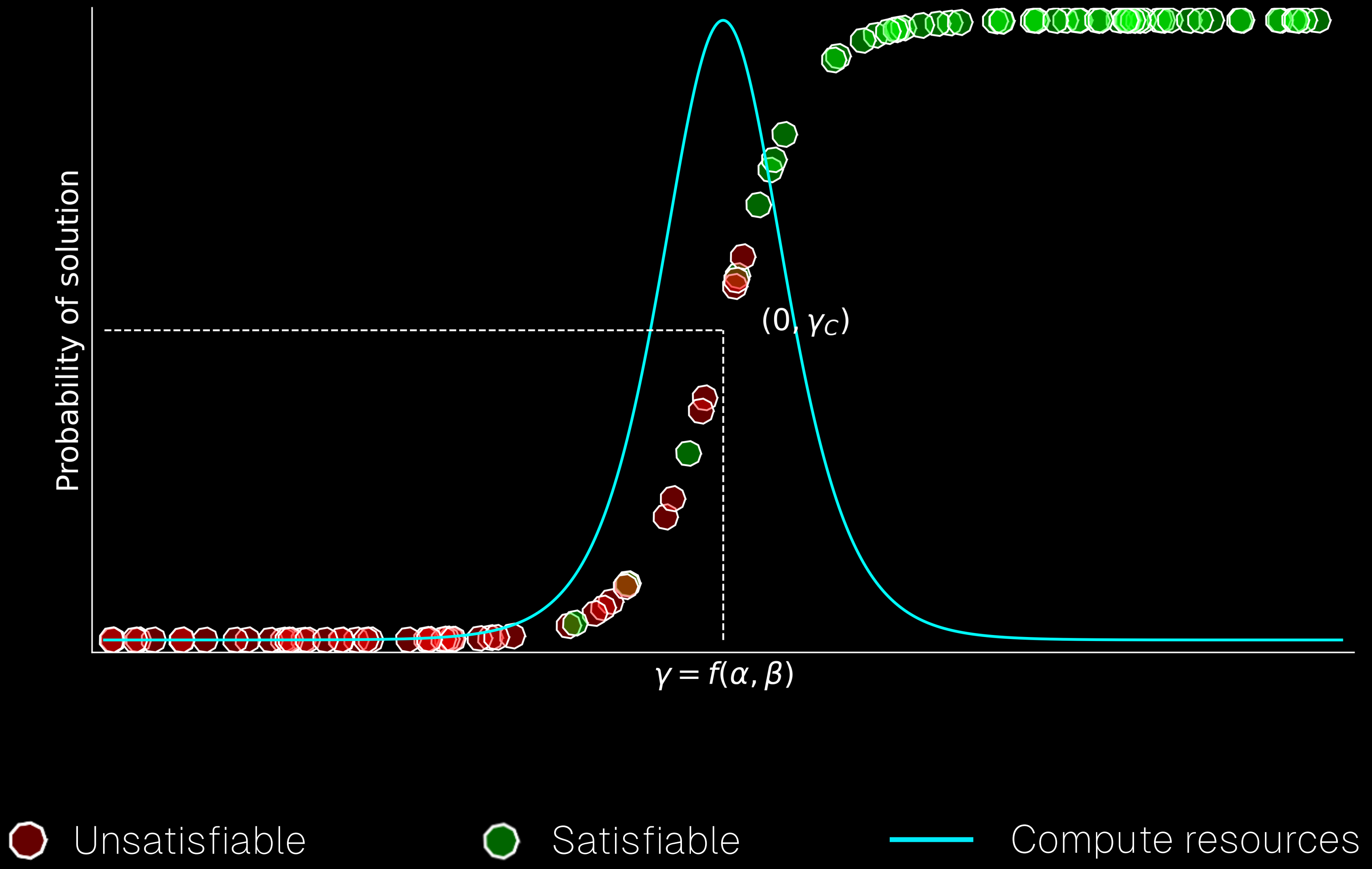
Reducing dimensions



Reducing dimensions



Reducing dimensions



Cheesman, et al. (1991)

Where the *Really* Hard Problems Are

Peter Cheeseman, RIACS*

Bob Kanefsky, Sterling Software

William M. Taylor, Sterling Software

Artificial Intelligence Research Branch
NASA Ames Research Center, Mail Stop 244-17
Moffett Field, CA 94035, USA
Email: < last-namO@ptolemy.arcnasa.gov

Abstract

It is well known that for many NP-complete problems, such as K-Sat, etc., typical cases are easy to solve; so that computationally hard cases must be rare (assuming P = NP). This paper shows that NP-complete problems can be summarized by at least one "order parameter", and that the hard problems occur at a critical value of such a parameter. This critical value separates two regions of characteristically different properties. For example, for K-colorability, the critical value separates overconstrained from underconstrained random graphs, and it marks the value at which the probability of a solution changes abruptly from near 0 to near 1. It is the high density of well separated almost solutions (local minima) at this boundary that cause search algorithms to "thrash". This boundary is a type of phase transition and we show that it is preserved under mappings between problems. We show that for some P problems either there is no phase transition or it occurs for bounded N (and so bounds the cost). These results suggest a way of deciding if a problem is in P or NP and why they are different.

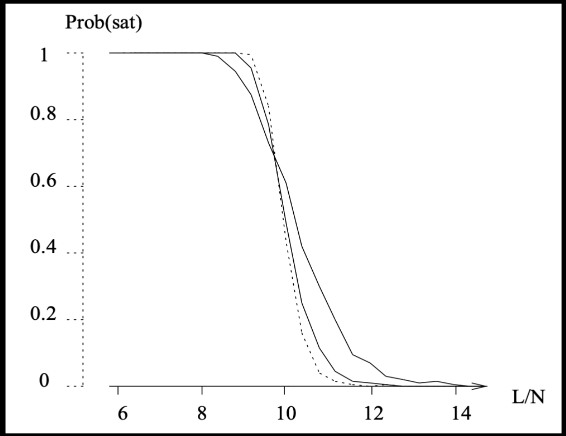
In this paper we show that for many NP problems one or more "order parameters" can be defined, and hard instances occur around particular critical values of these order parameters. In addition, such critical values form a boundary that separates the space of problems into two regions. One region is underconstrained, so the density of solutions is high, thus making it relatively easy to find a solution. The other region is overconstrained and very unlikely to contain a solution. If there are solutions in this overconstrained region, then they have such deep local minimum (strong basin of attraction) that any reasonable algorithm is likely to find it. If there is no solution, then a backtrack search can usually establish this with ease, since potential solution paths are usually cut off early in the search. Really hard problems occur on the boundary between these two regions, where the probability of a solution is low but non-negligible. At this point there are typically many local minima corresponding to almost solutions separated by high "energy barriers". These almost solutions form deep local minima that may often trap search methods that rely on local information.

Because it is possible to locate a region where hard problems occur, it is possible to predict whether a particular problem is likely to be easy to solve. We expect that in future computer scientists will produce "phase diagrams" for particular problem domains to aid in hard problem identification and for prediction of solution existence probability, such as shown in [6].

Cheesman, et al. (1991)

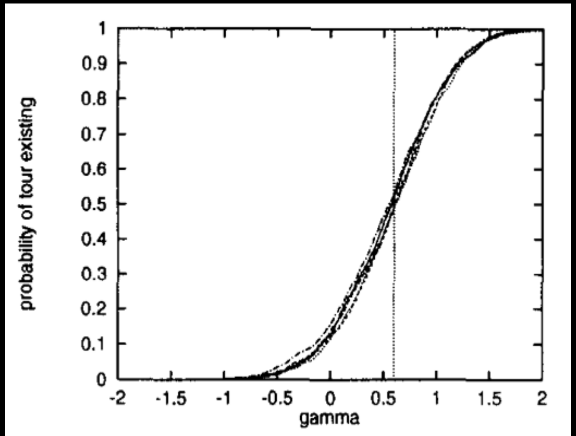
K-SAT

Gent & Walsh (1994)



TSP

Gent & Walsh (1996)



Where the *Really* Hard Problems Are

Peter Cheeseman — RIACS*

Bob Kanefsky — Sterling Software

William M. Taylor — Sterling Software

Artificial Intelligence Research Branch
NASA Ames Research Center, Mail Stop 244-17
Moffett Field, CA 94035, USA
Email: < last-namO@ptolemy.arcnasa.gov

Abstract

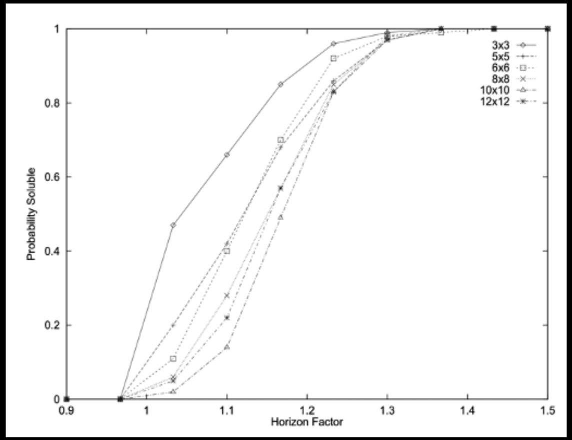
It is well known that for many NP-complete problems, such as K-Sat, etc., typical cases are easy to solve; so that computationally hard cases must be rare (assuming P = NP). This paper shows that NP-complete problems can be summarized by at least one "order parameter", and that the hard problems occur at a critical value of such a parameter. This critical value separates two regions of characteristically different properties. For example, for K-colorability, the critical value separates overconstrained from underconstrained random graphs, and it marks the value at which the probability of a solution changes abruptly from near 0 to near 1. It is the high density of well separated almost solutions (local minima) at this boundary that cause search algorithms to "thrash". This boundary is a type of phase transition and we show that it is preserved under mappings between problems. We show that for some P problems either there is no phase transition or it occurs for bounded N (and so bounds the cost). These results suggest a way of deciding if a problem is in P or NP and why they are different.

In this paper we show that for many NP problems one or more "order parameters" can be defined, and hard instances occur around particular critical values of these order parameters. In addition, such critical values form a boundary that separates the space of problems into two regions. One region is underconstrained, so the density of solutions is high, thus making it relatively easy to find a solution. The other region is overconstrained and very unlikely to contain a solution. If there are solutions in this overconstrained region, then they have such deep local minimum (strong basin of attraction) that any reasonable algorithm is likely to find it. If there is no solution, then a backtrack search can usually establish this with ease, since potential solution paths are usually cut off early in the search. Really hard problems occur on the boundary between these two regions, where the probability of a solution is low but non-negligible. At this point there are typically many local minima corresponding to almost solutions separated by high "energy barriers". These almost solutions form deep local minima that may often trap search methods that rely on local information.

Because it is possible to locate a region where hard problems occur, it is possible to predict whether a particular problem is likely to be easy to solve. We expect that in future computer scientists will produce "phase diagrams" for particular problem domains to aid in hard problem identification and for prediction of solution existence probability, such as shown in [6].

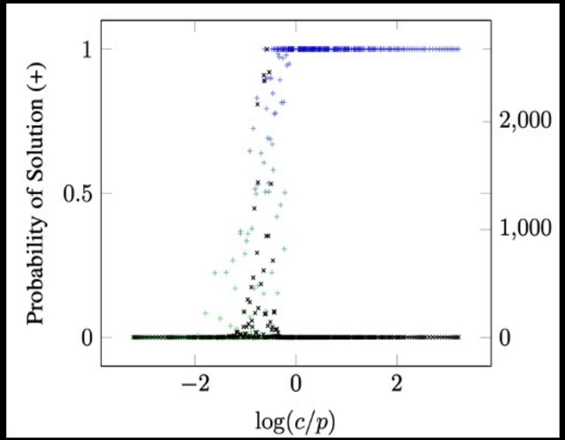
Job scheduling

Beck & Jackson (1997)



Knapsack

Yadav, et al. (2020)



1. Phase transitions
2. Knapsack (decision)
3. Knapsack (optimisation)

Knapsack decision problem

Input:

- $W = \{w_1, \ldots, w_n\}$, weight vector
- $P = \{p_1, \ldots, p_n\}$, profit vector
- C = capacity constraint
- T = threshold

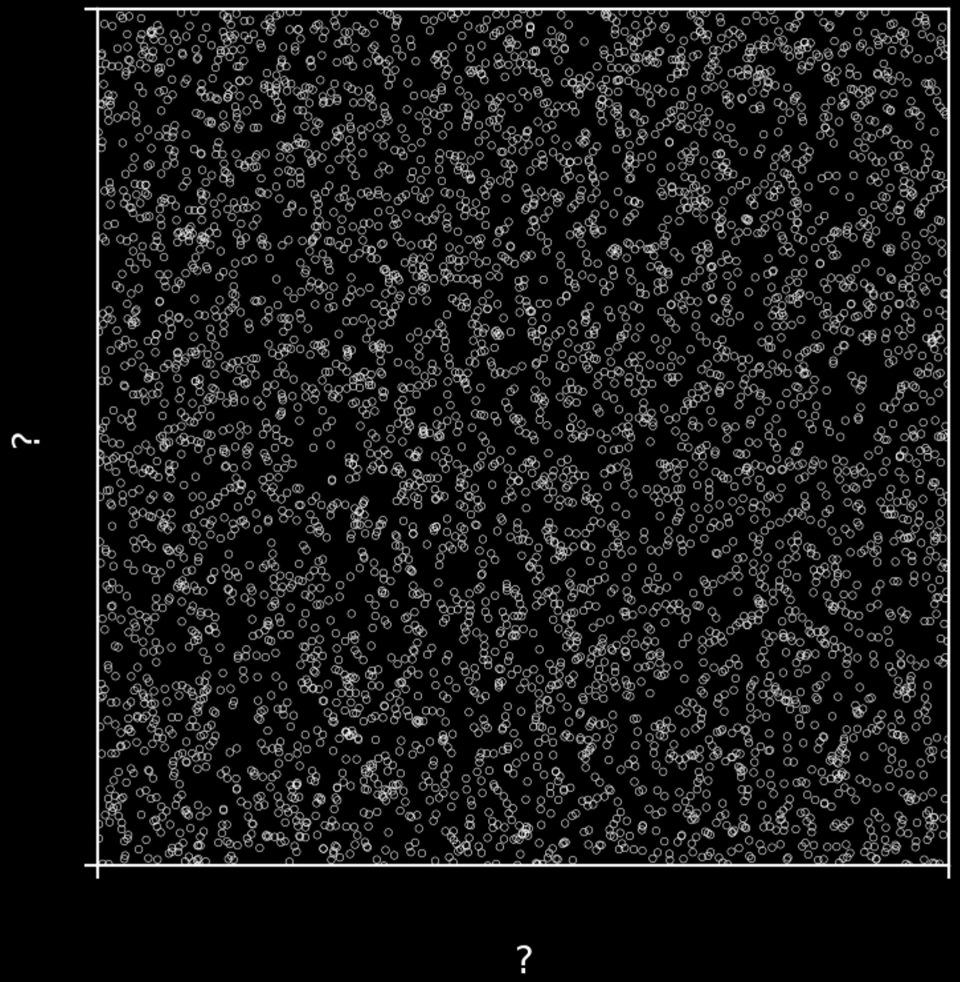
Output:

'TRUE' if a satisfying assignment $X = \{x_1, \ldots, x_n\}$, $x_i \in \{0, 1\}$ exists, such that:

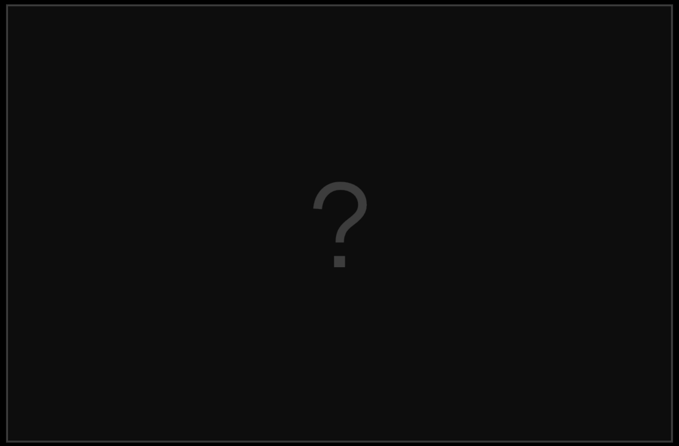
$$\sum_{i=1}^{n} w_i x_i \leq C \quad \text{and} \quad \sum_{i=1}^{n} p_i x_i \geq T,$$

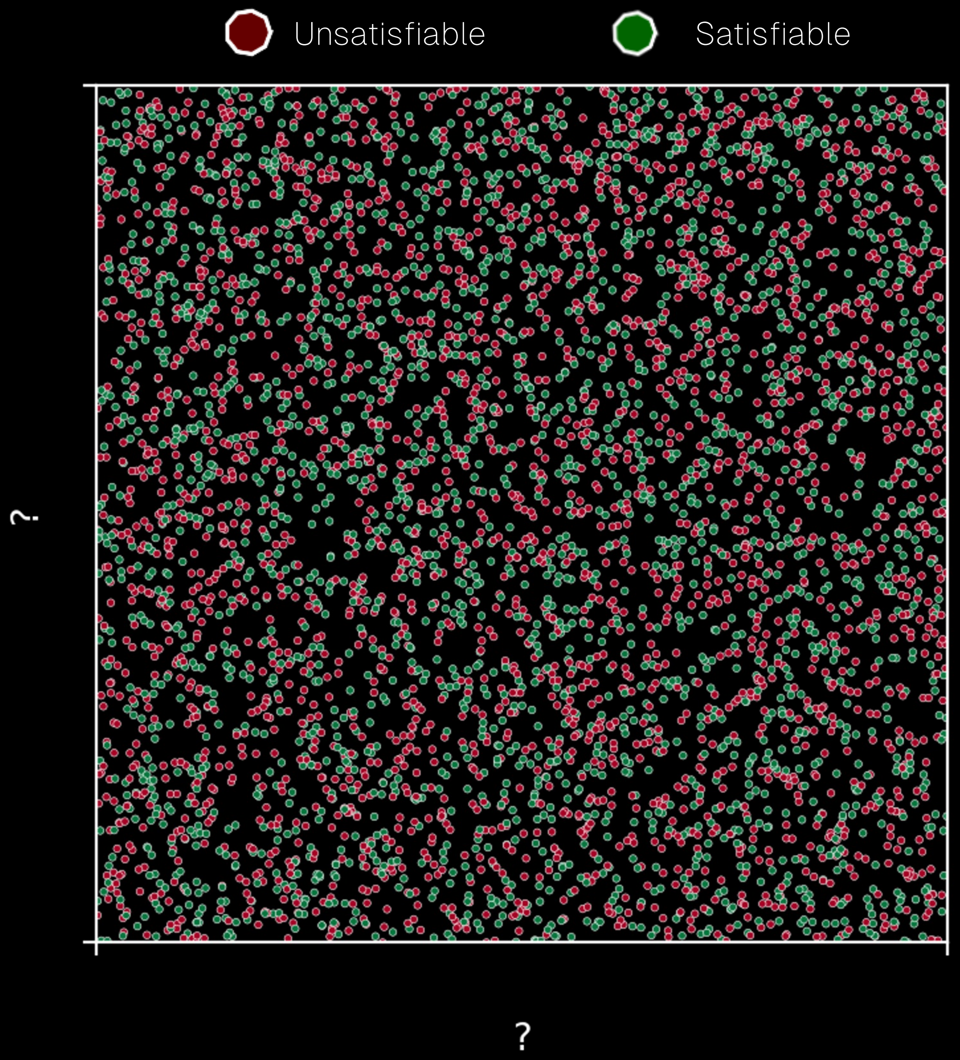
or 'FALSE' if no such assignment exists.

Setup

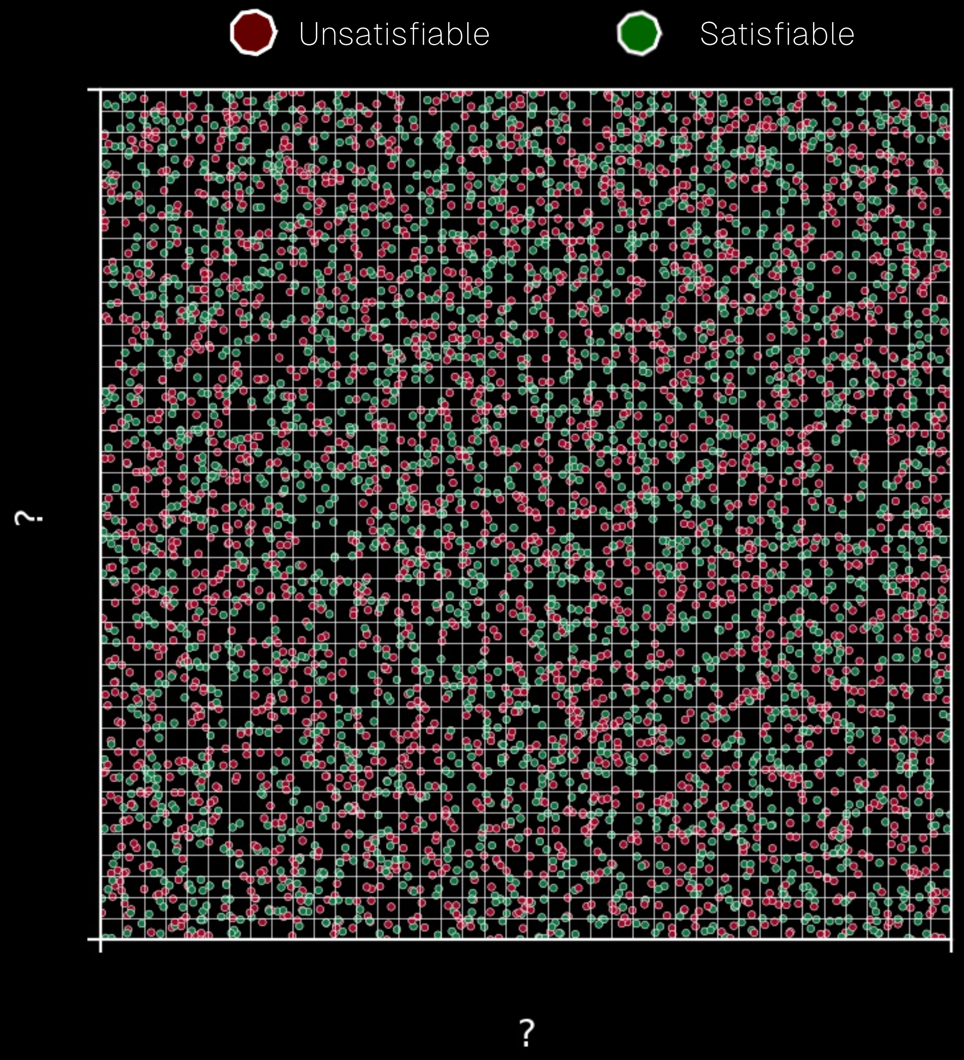


Setup

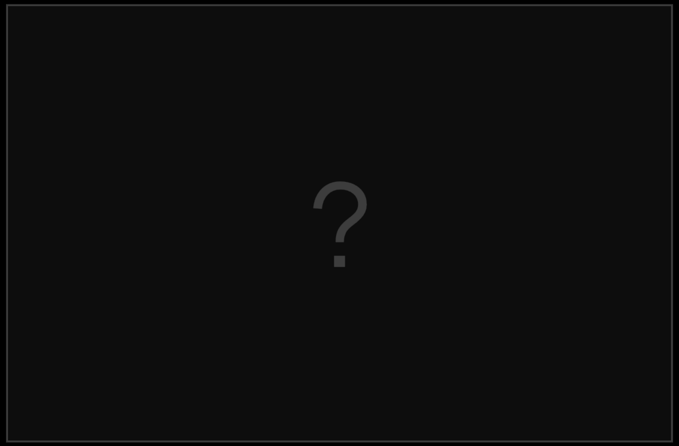


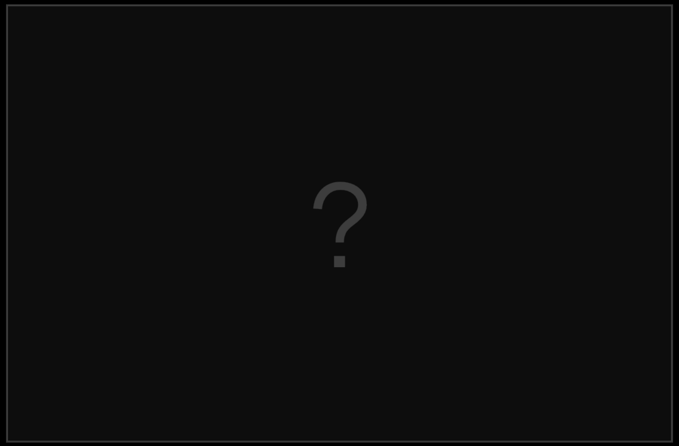


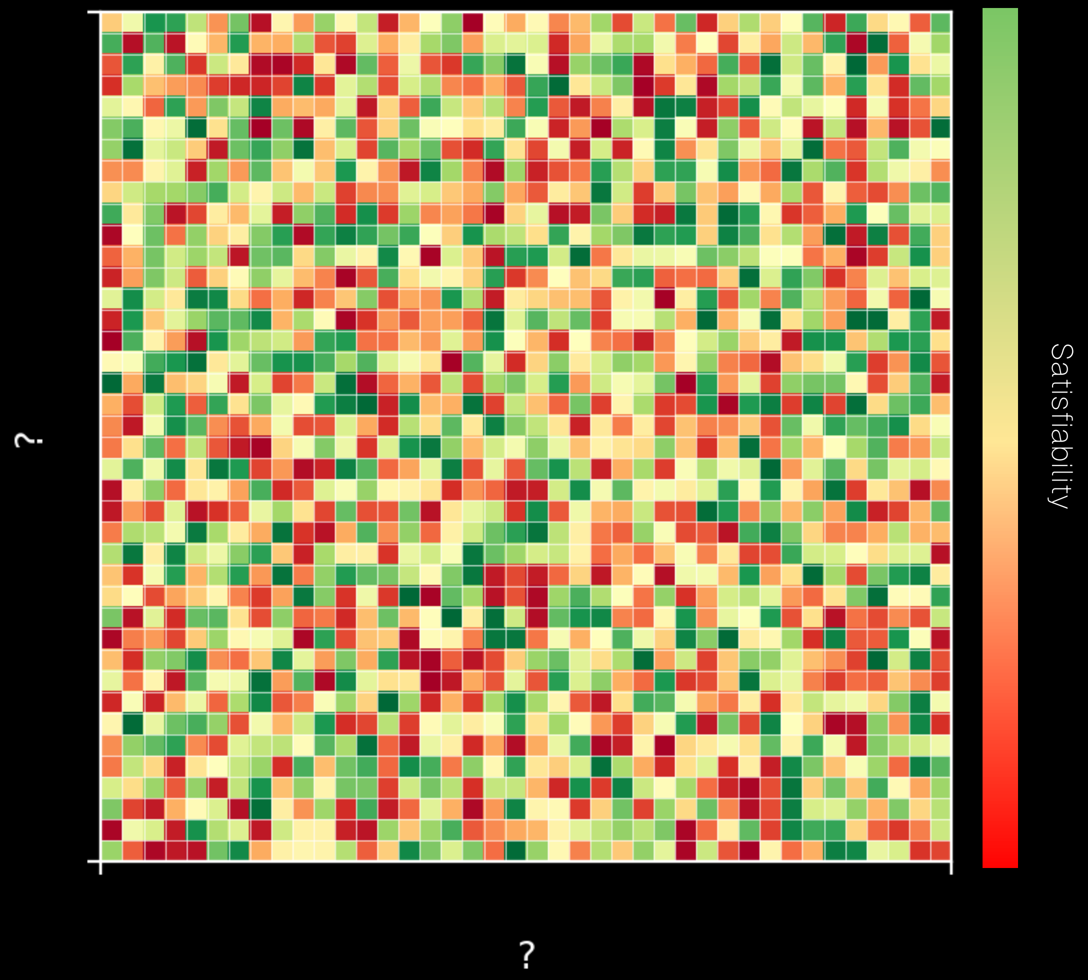
Setup



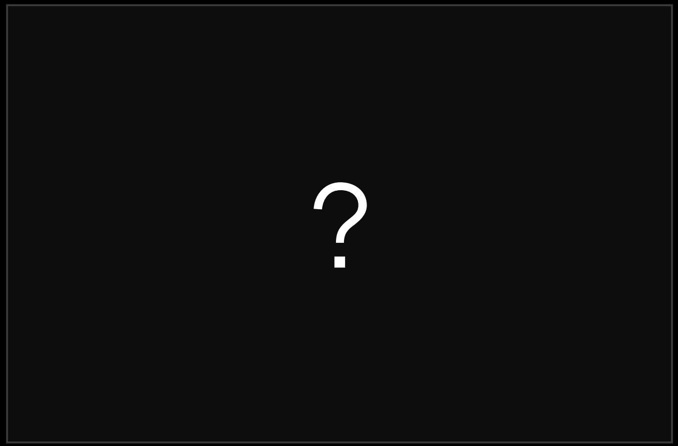
Setup

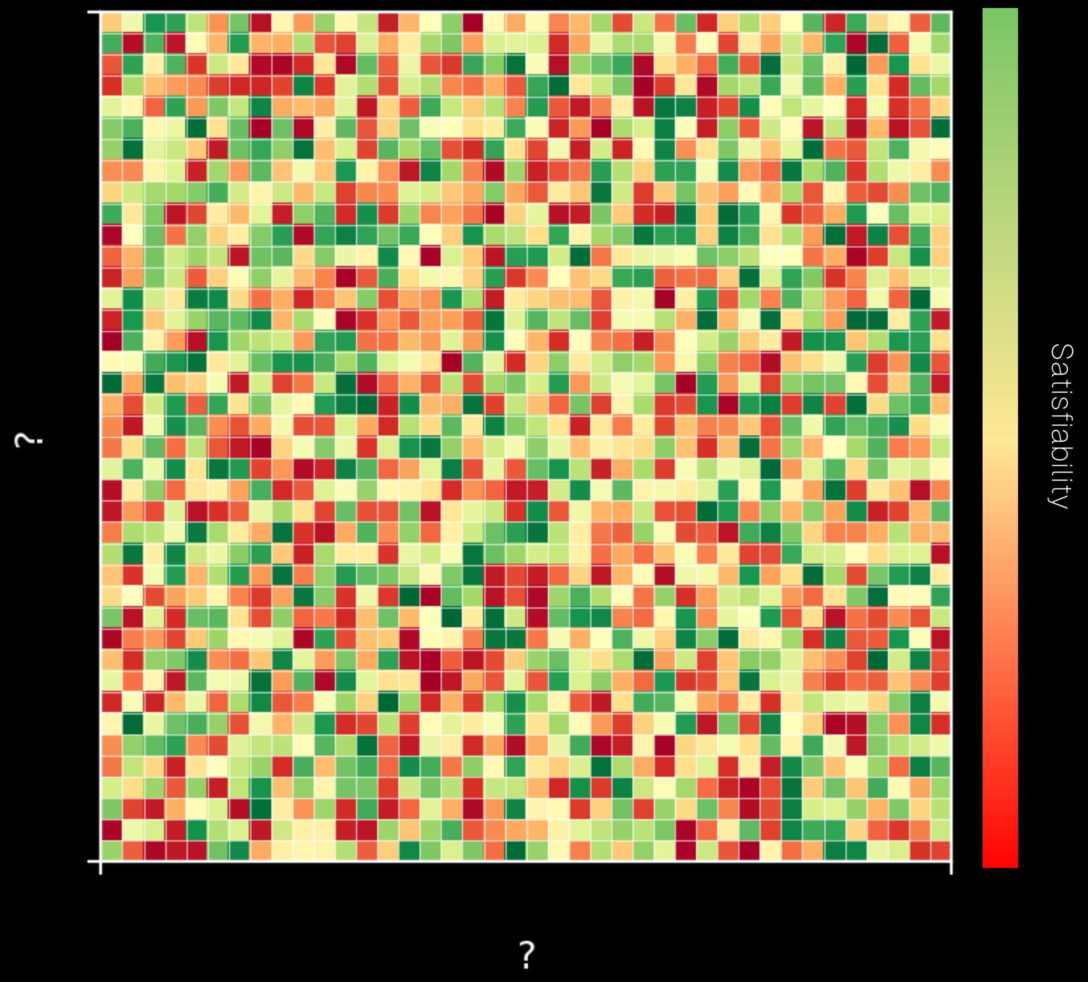






Order parameters





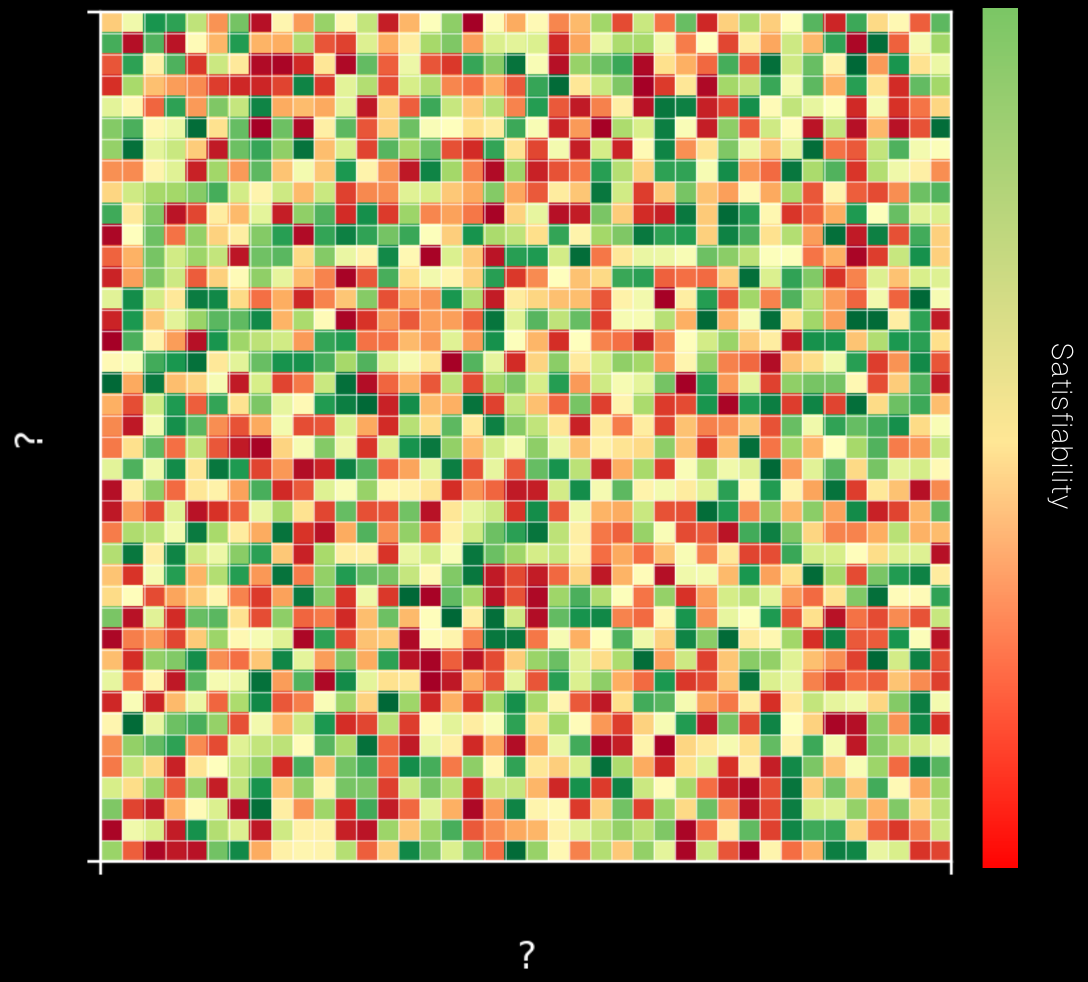
Order parameters

(1) Normalised capacity

Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

?



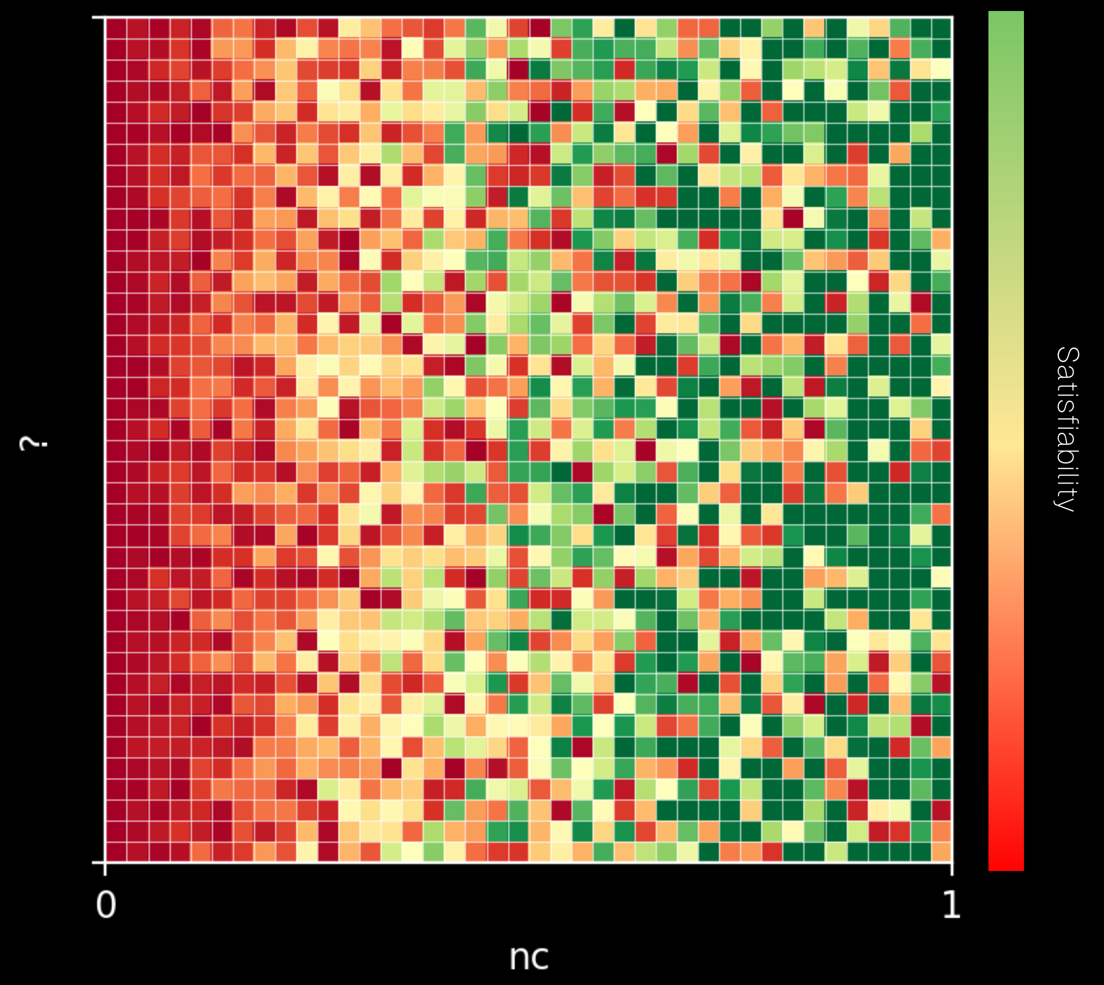
Order parameters

(1) Normalised capacity

Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

?



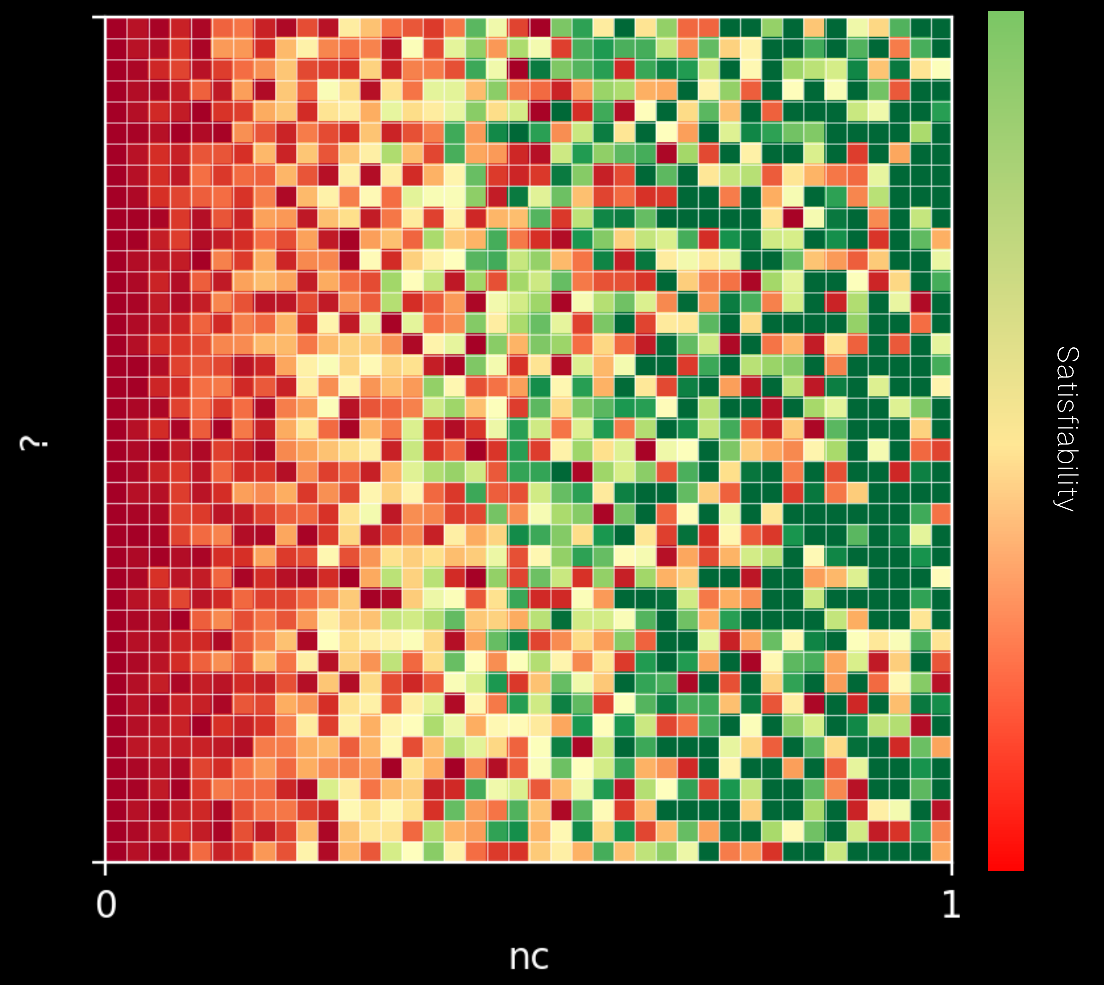
Order parameters

(1) Normalised capacity

Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

?



Order parameters

(1) Normalised capacity

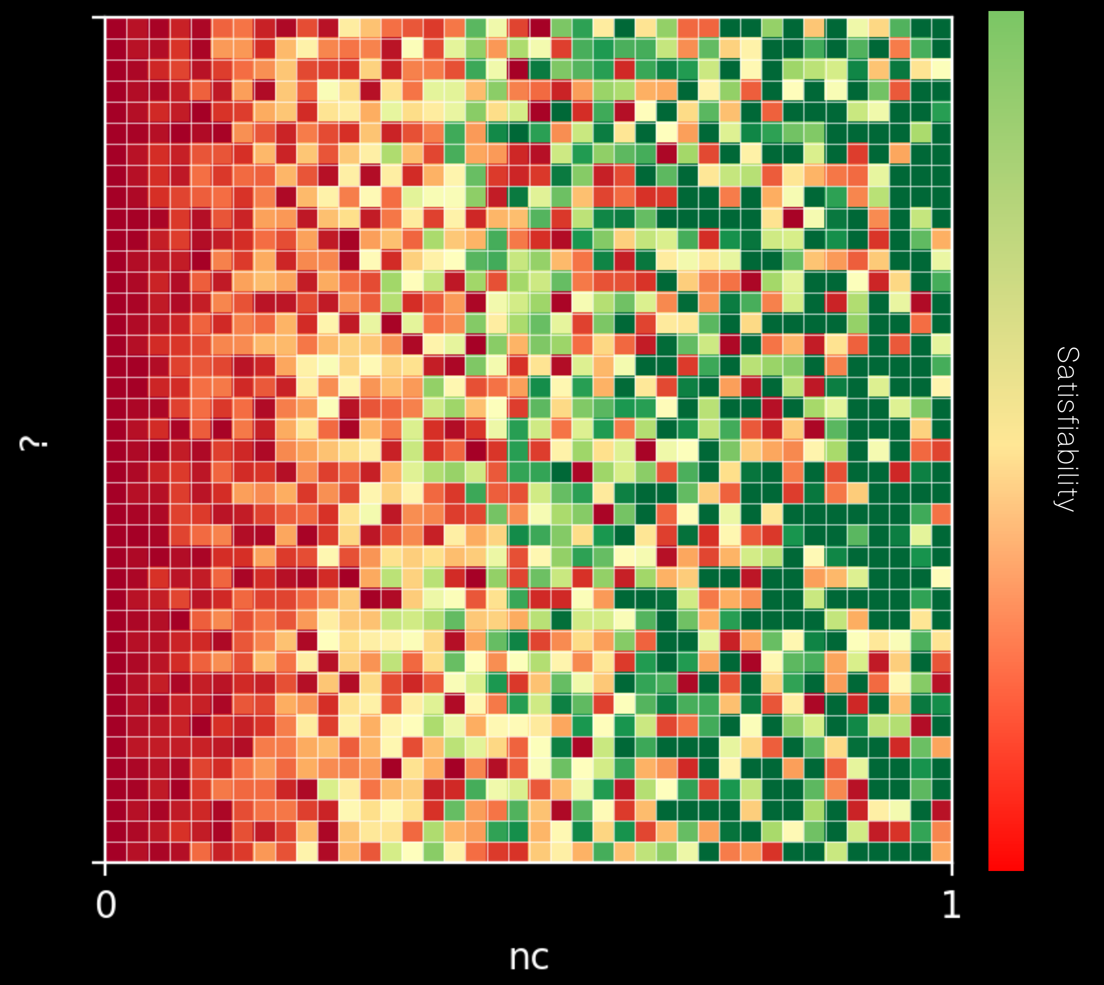
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Phase transition

(1) Normalised capacity

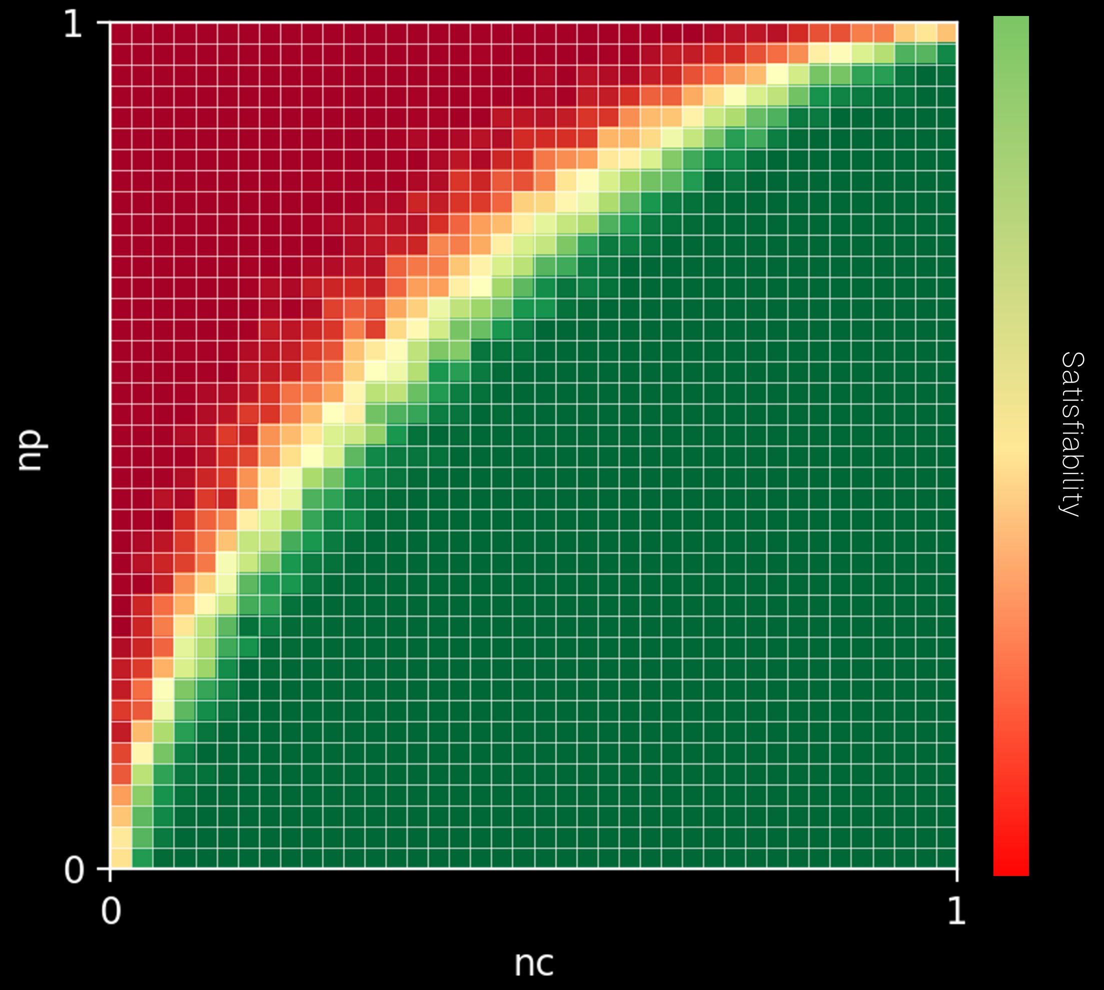
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Constrainedness

(1) Normalised capacity

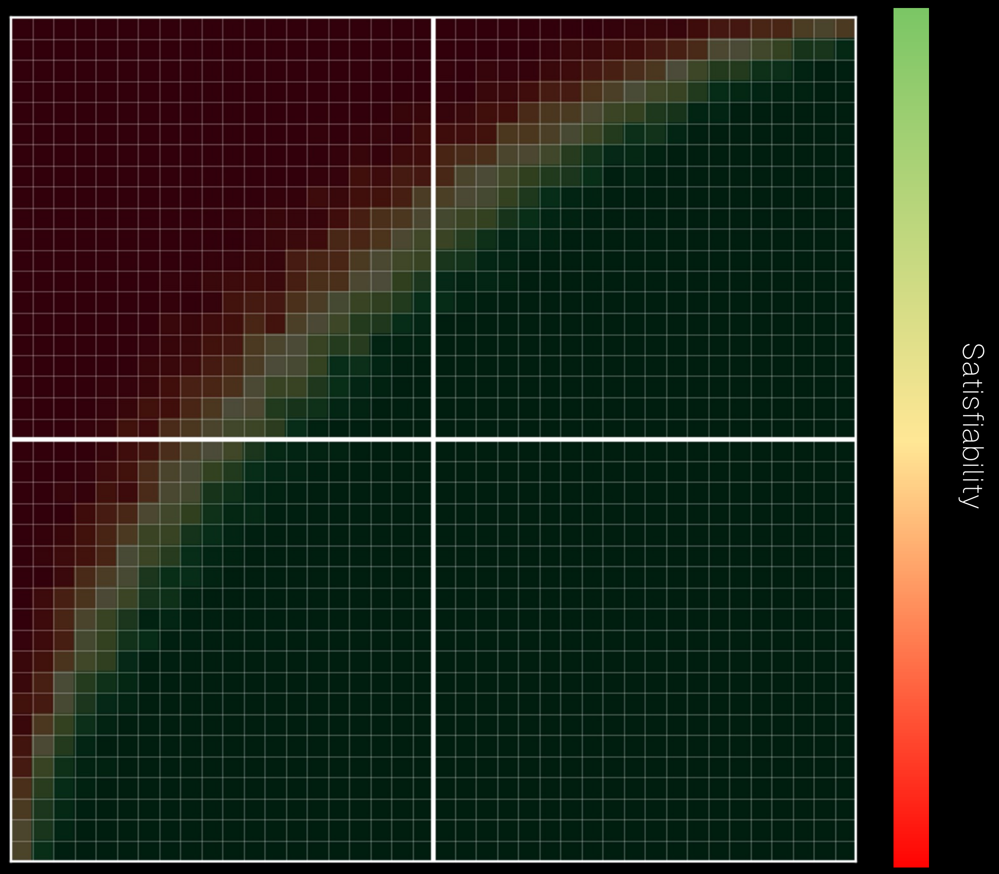
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Constrainedness

(1) Normalised capacity

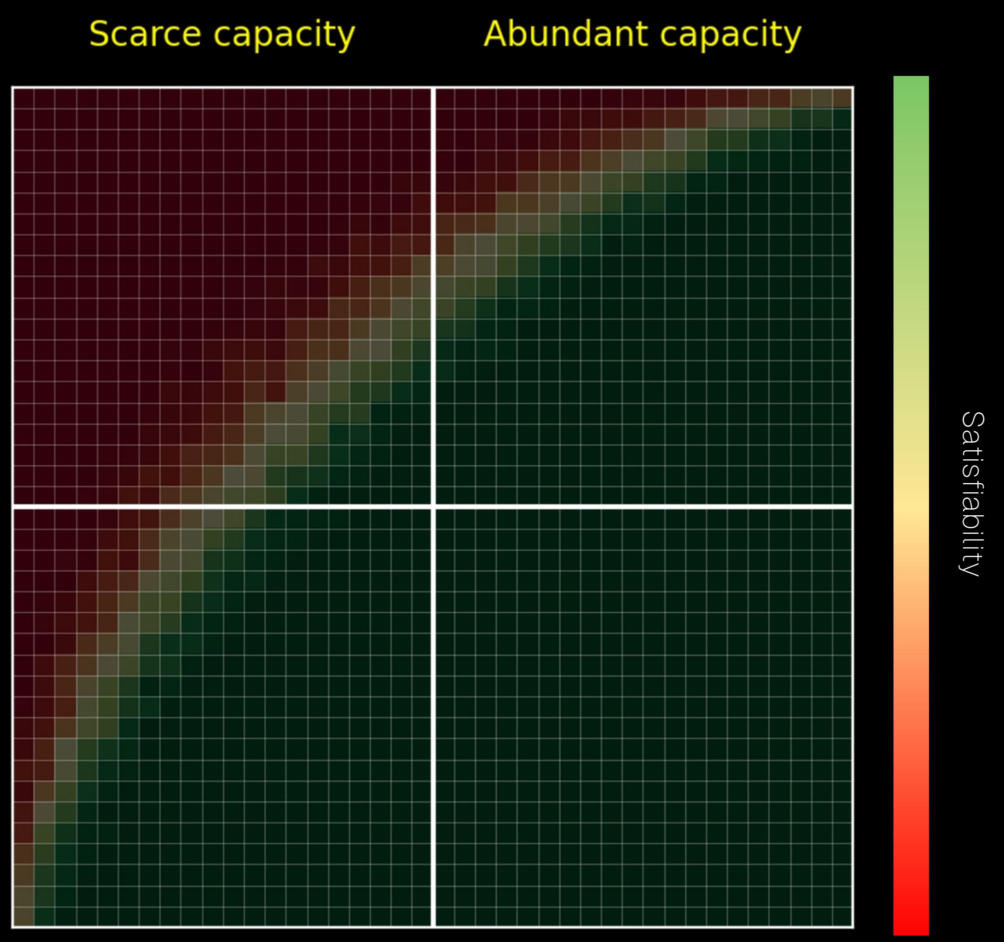
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Constrainedness

(1) Normalised capacity

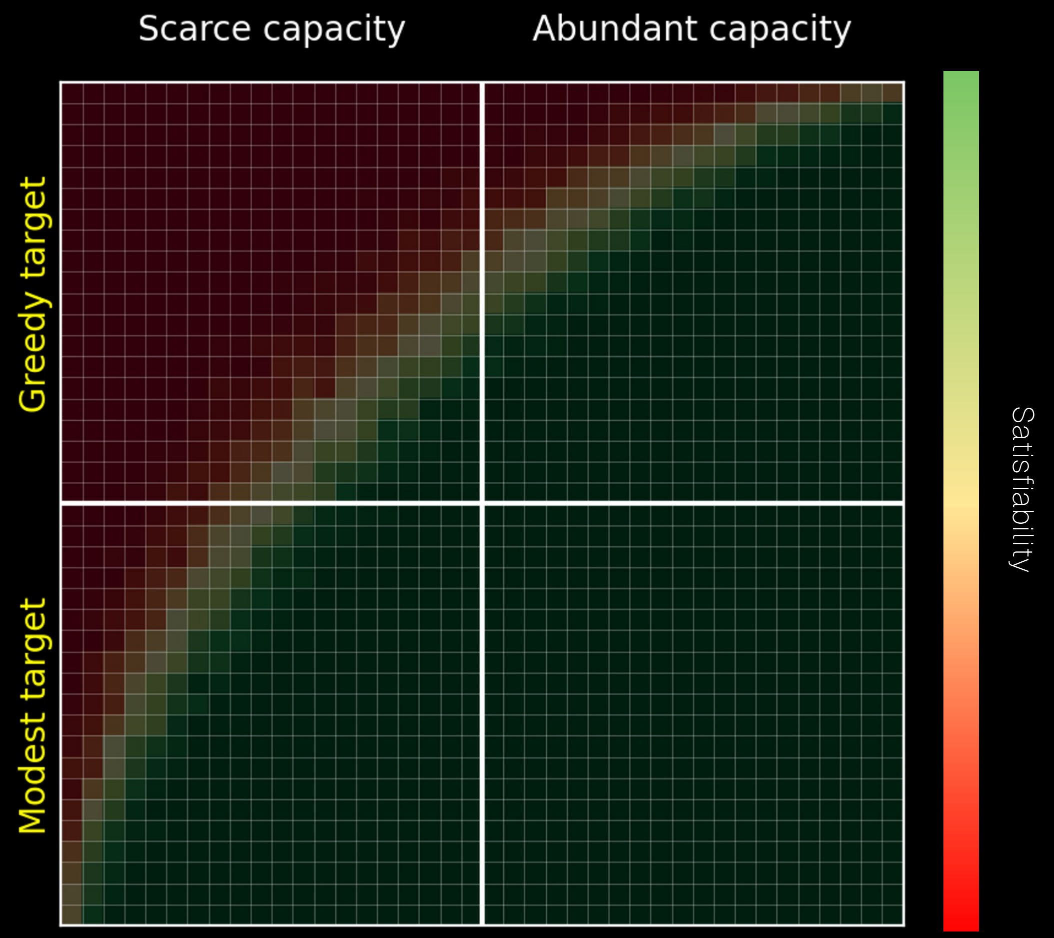
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Constrainedness

(1) Normalised capacity

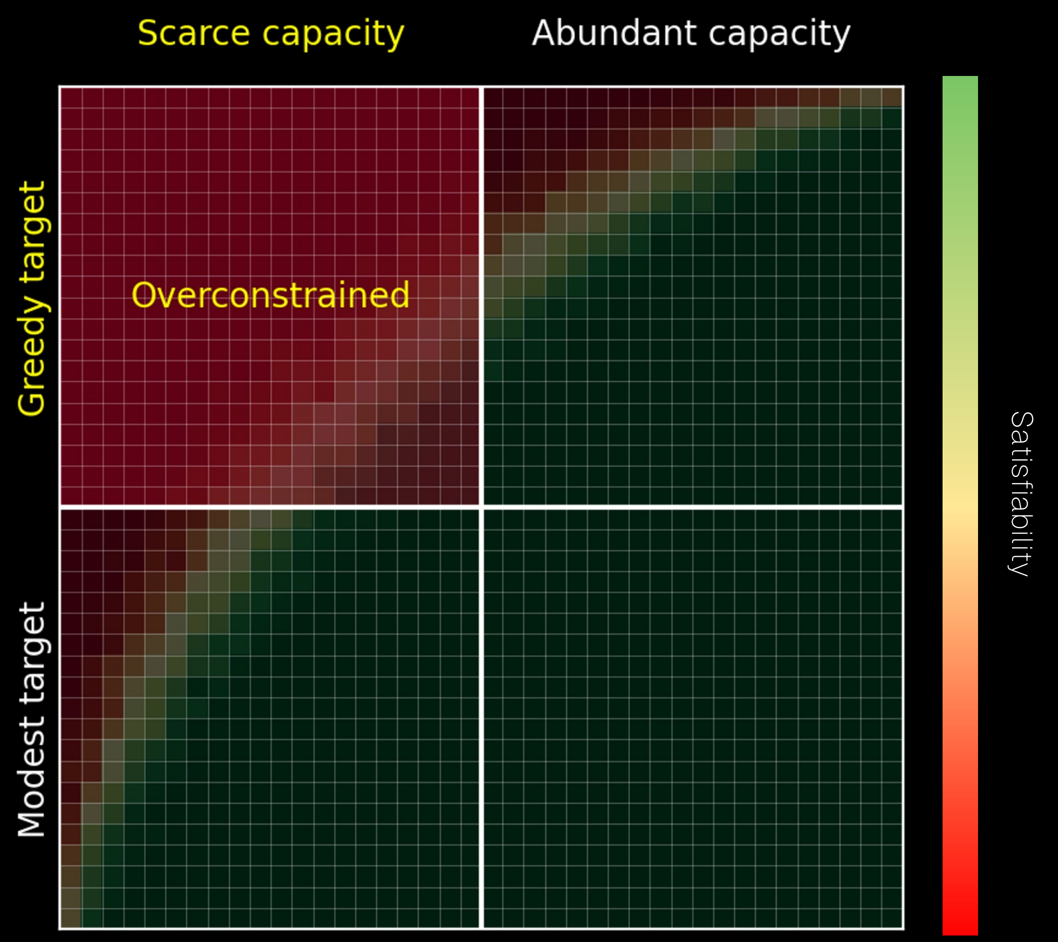
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Constrainedness

(1) Normalised capacity

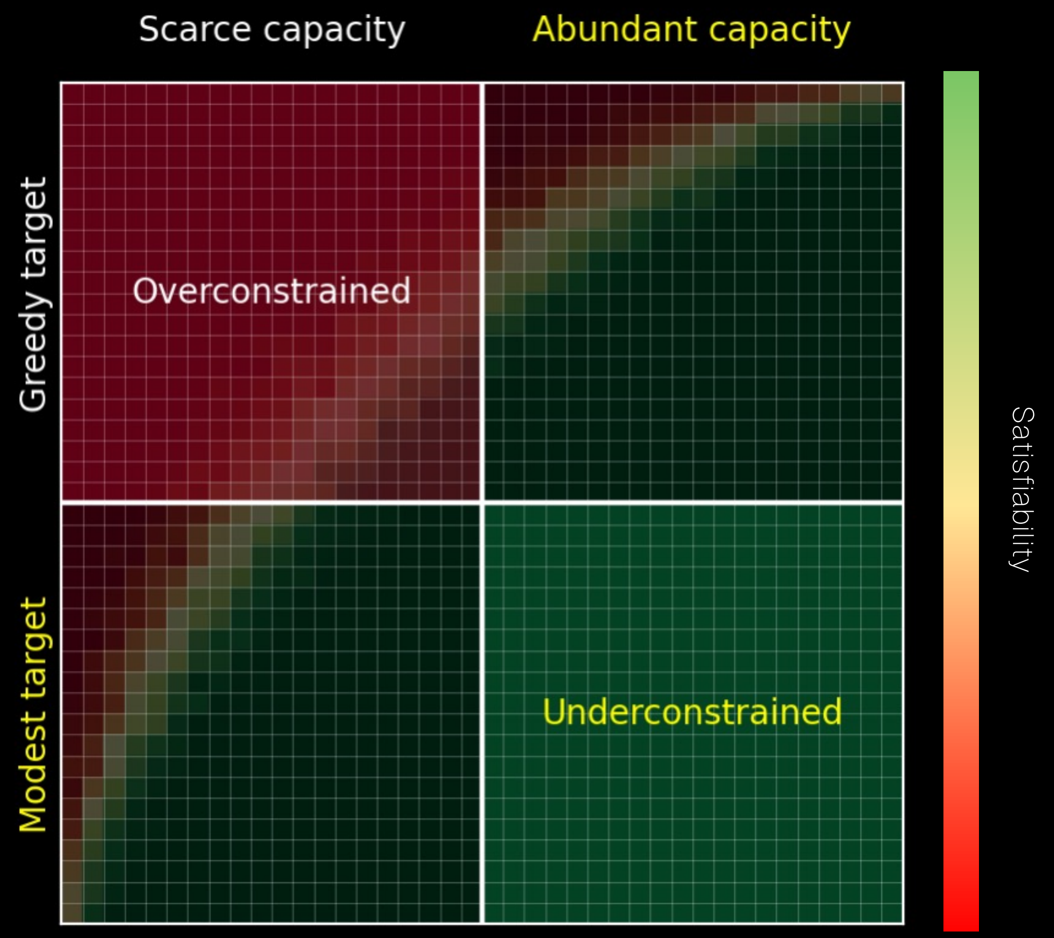
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Constrainedness

(1) Normalised capacity

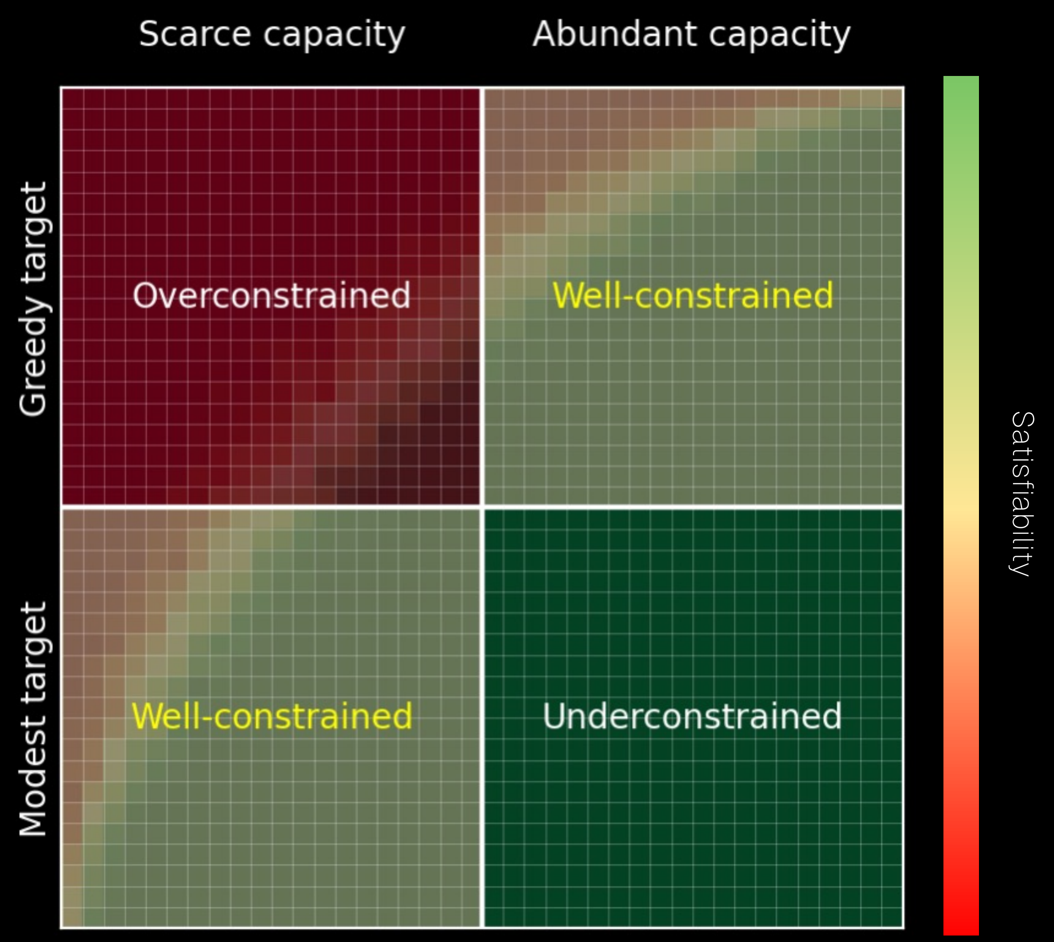
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

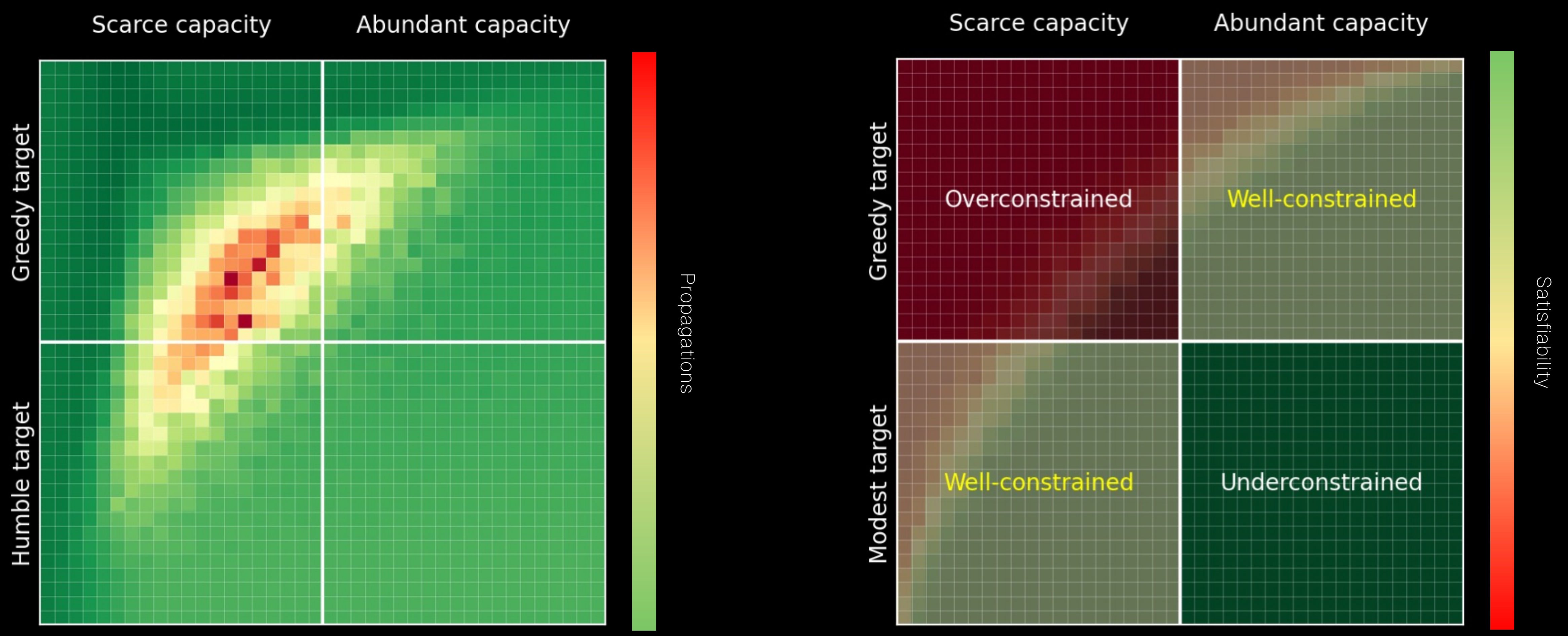
(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Constrainedness



Reducing dimensions

f (

(1) Normalised capacity

Abundance of capacity

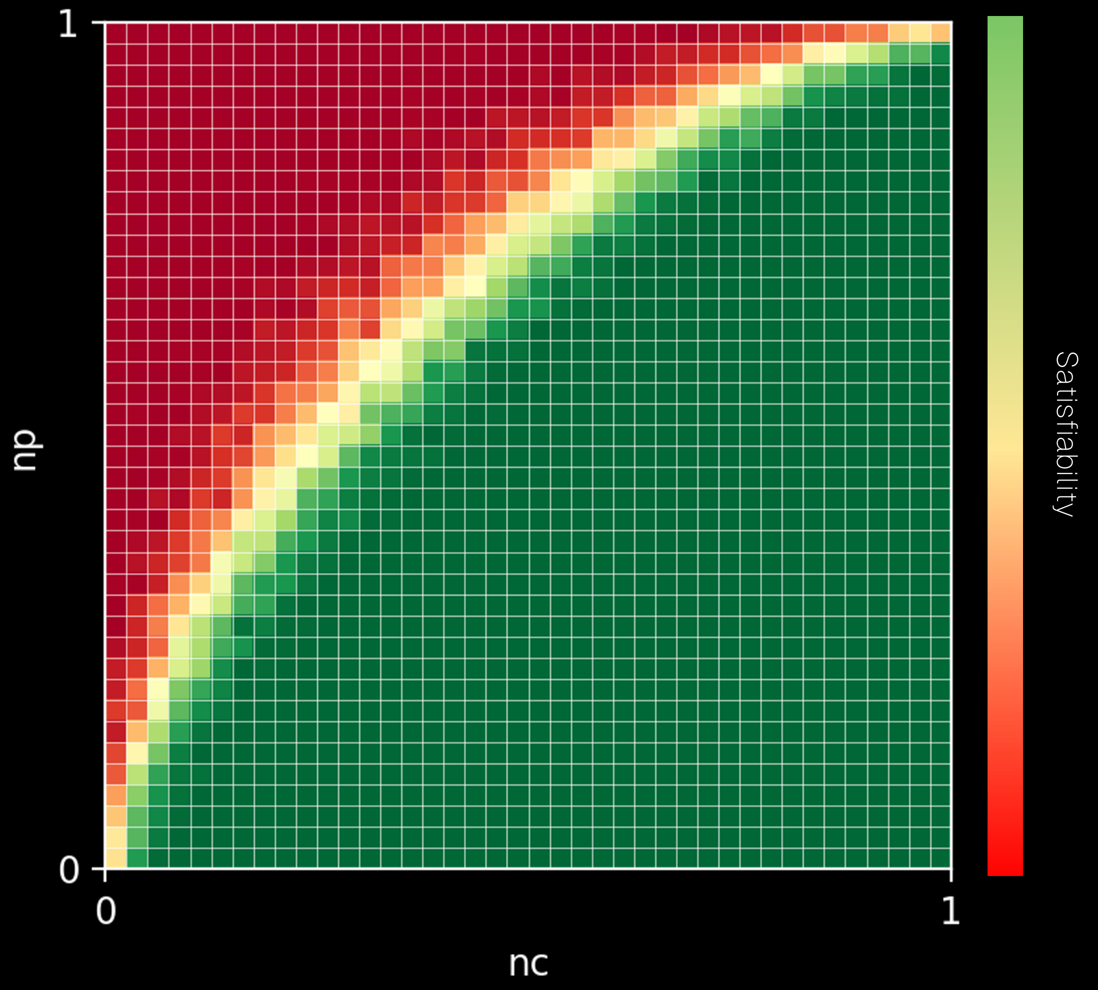
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

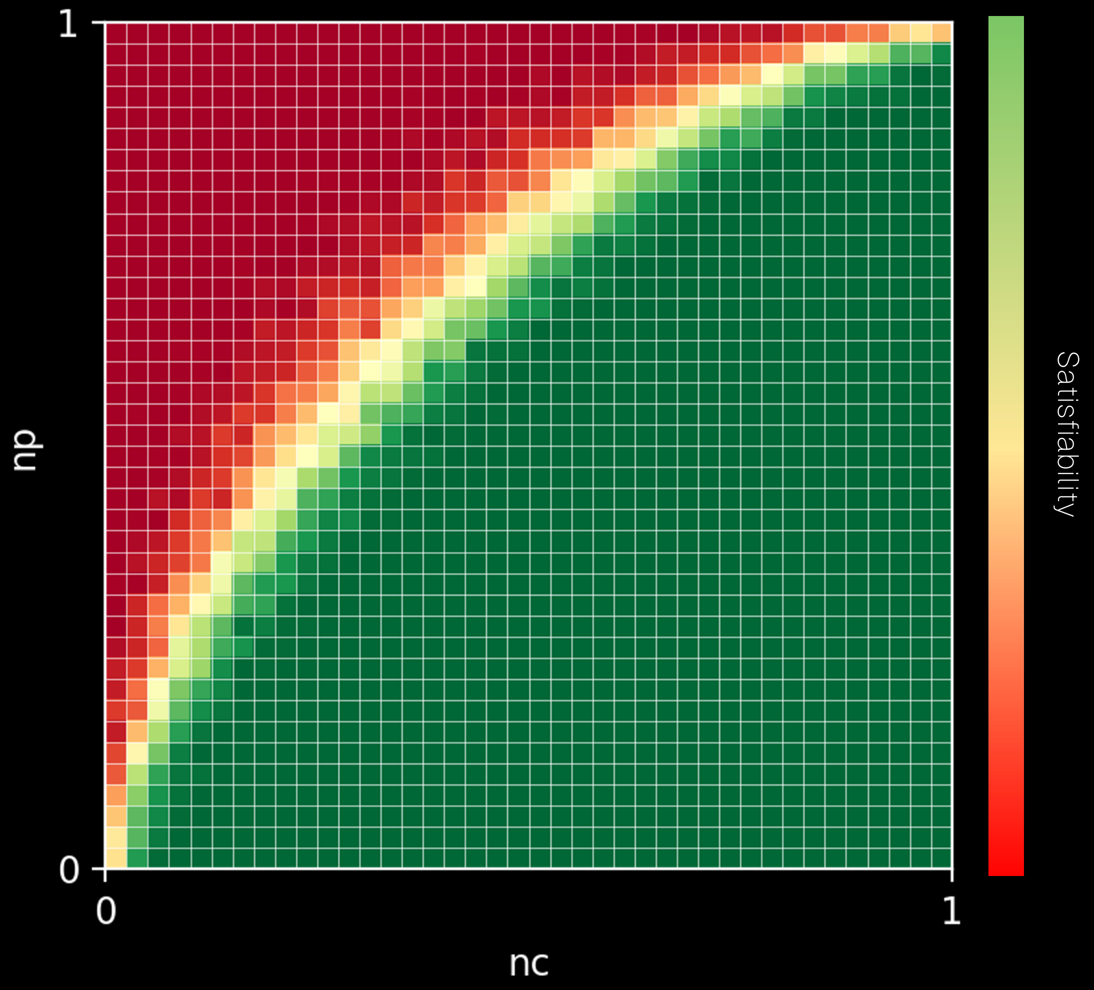
$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$

) $= \gamma$



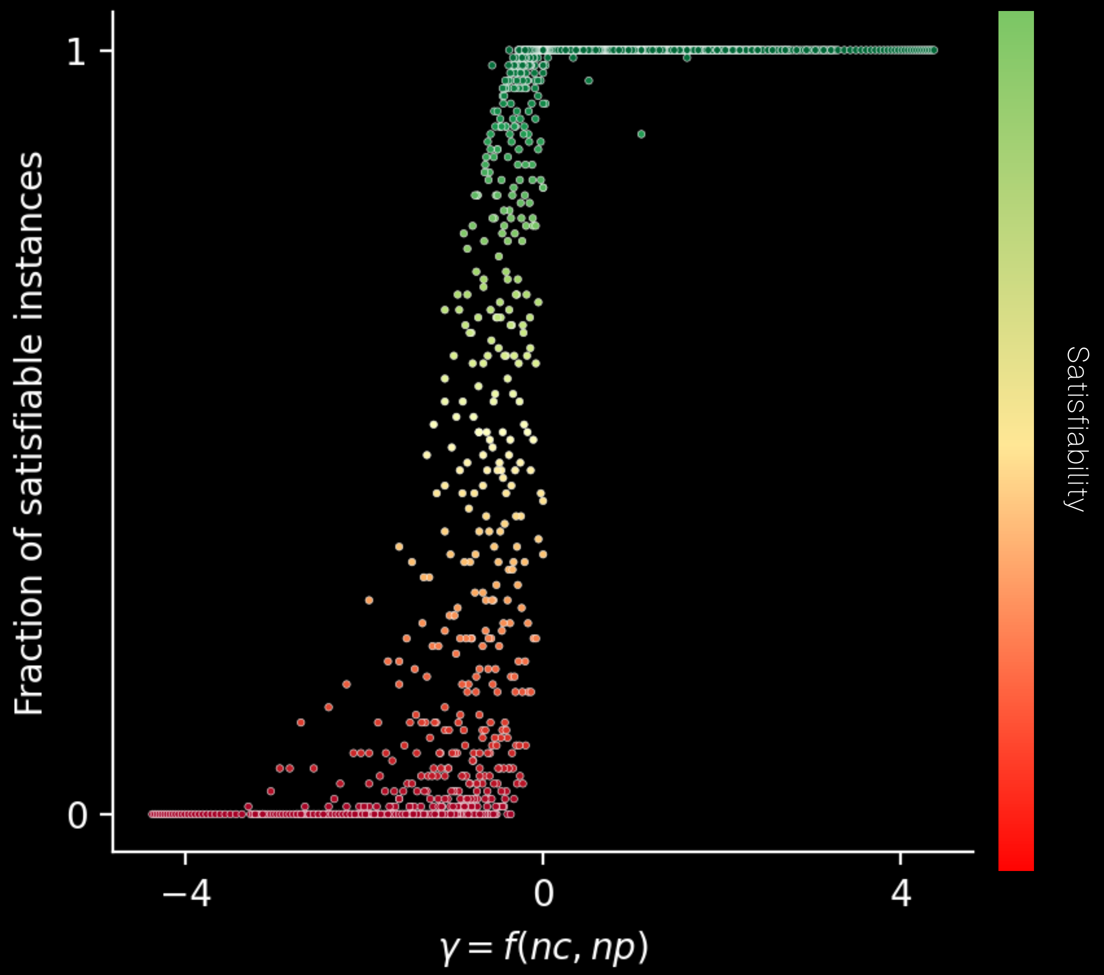
Reducing dimensions

$$\log\left(\frac{nc}{np}\right) = \gamma$$



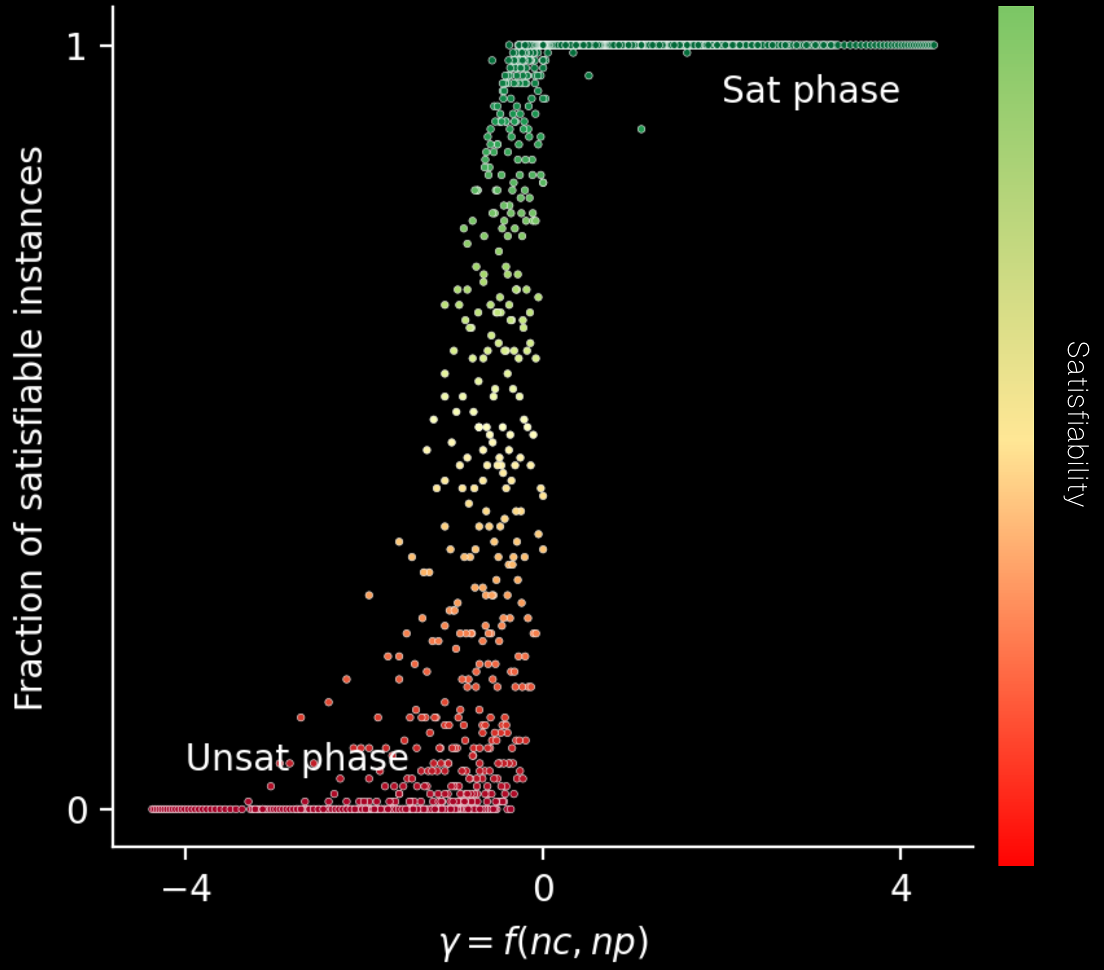
Reducing dimensions

$$\log\left(\frac{nc}{np}\right) = \gamma$$

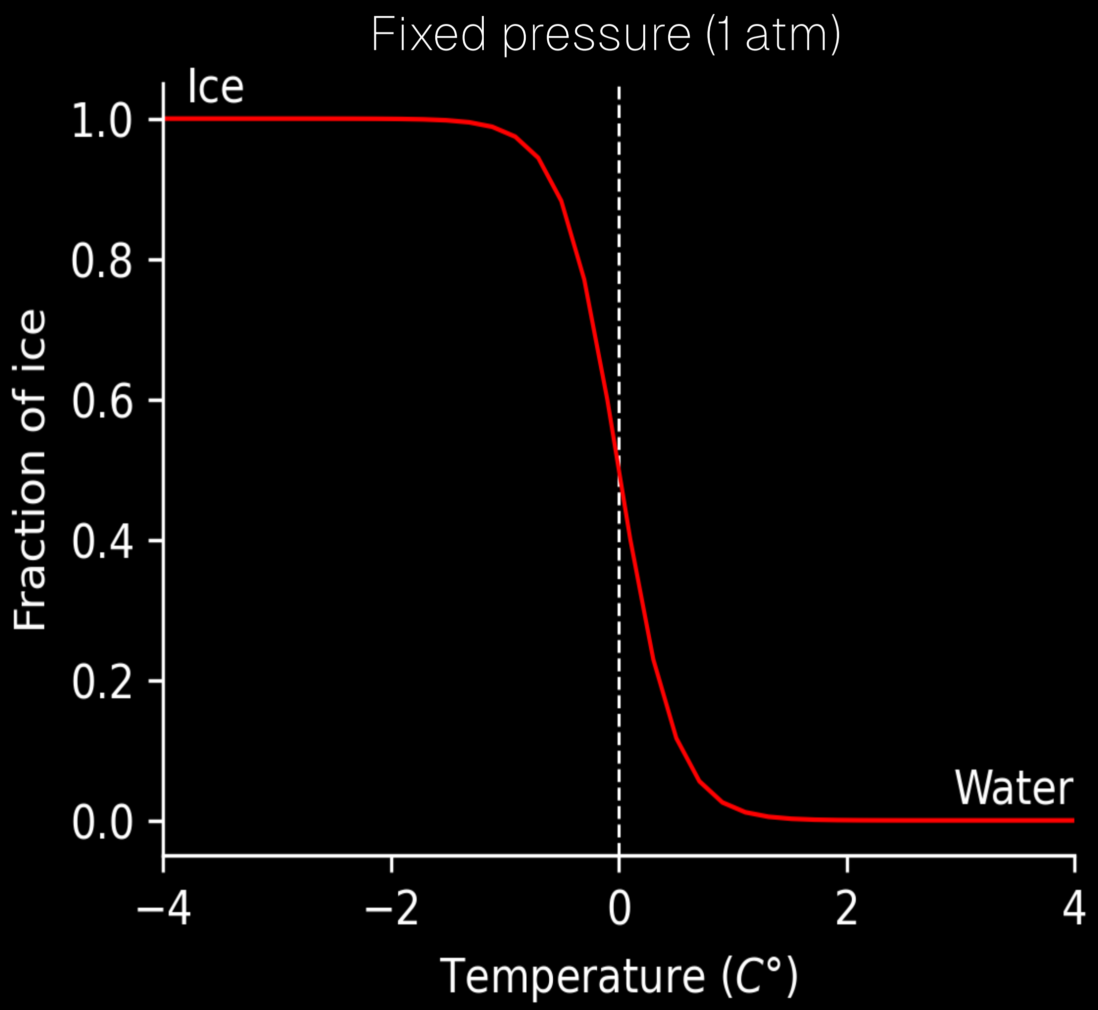


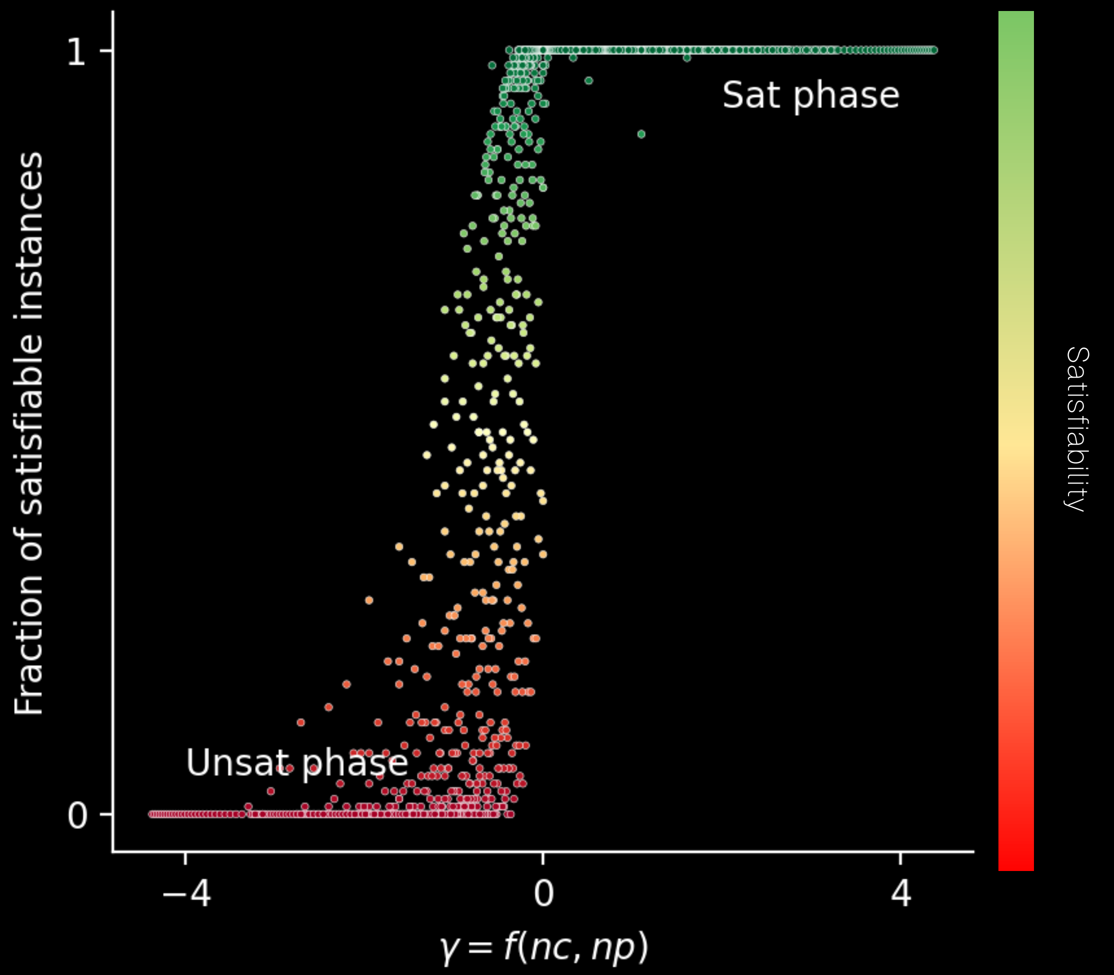
Reducing dimensions

$$\log\left(\frac{nc}{np}\right) = \gamma$$



Reducing dimensions





1. Phase transitions
2. Knapsack (decision)
3. Knapsack (optimisation)

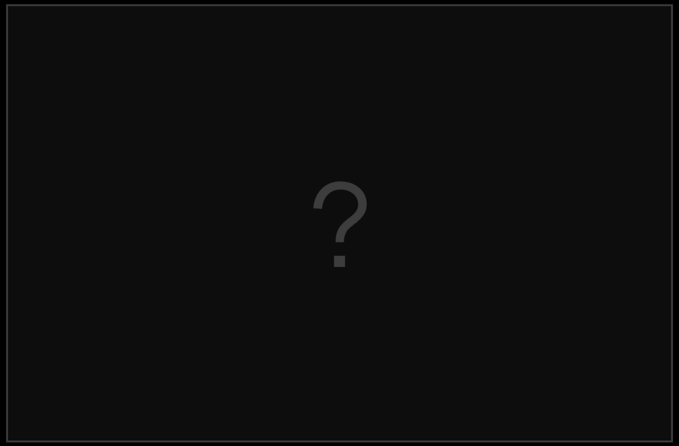
What about optimisation problems?

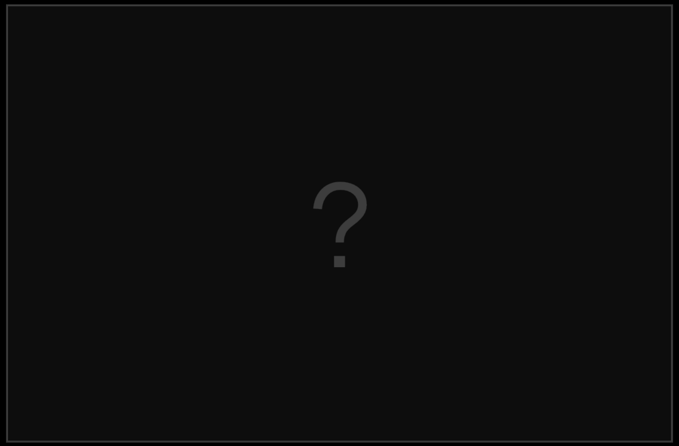
Sensible phases?

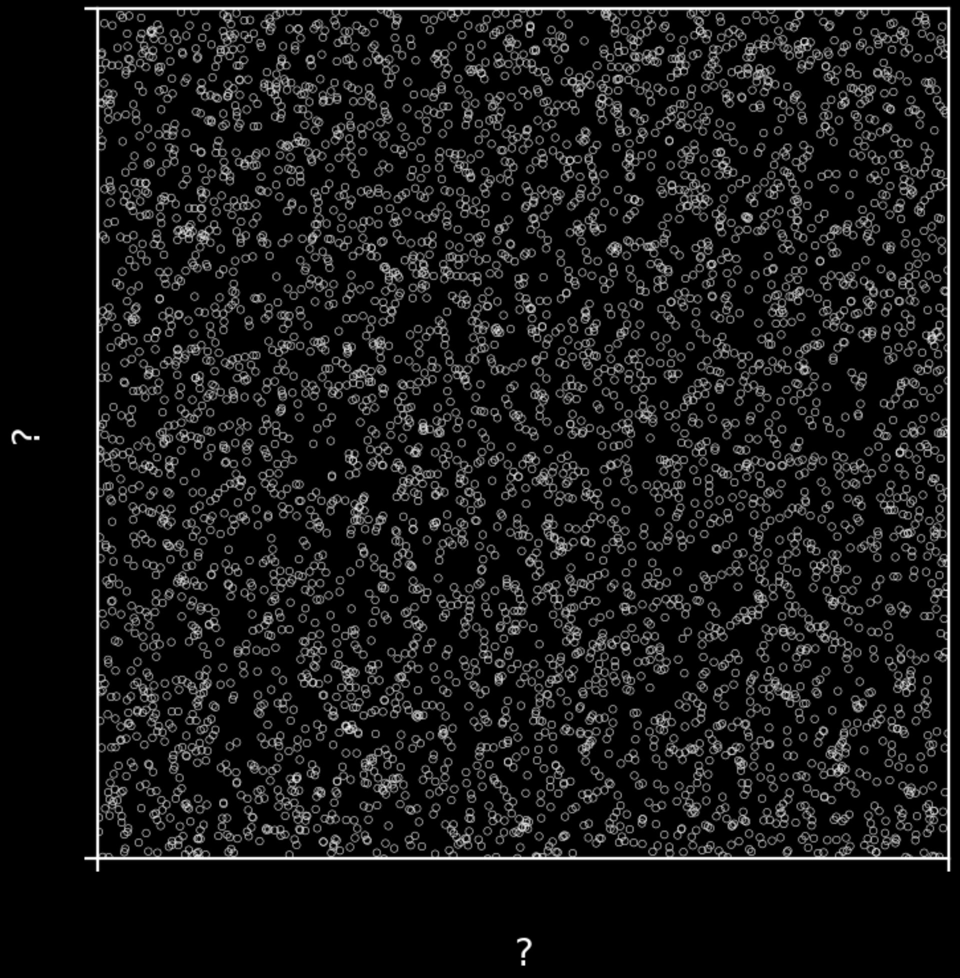
● ~~Unsatisfiable~~ ● ~~Satisfiable~~

Sensible order parameter(s)?

Setup







Setup

(1) Normalised capacity

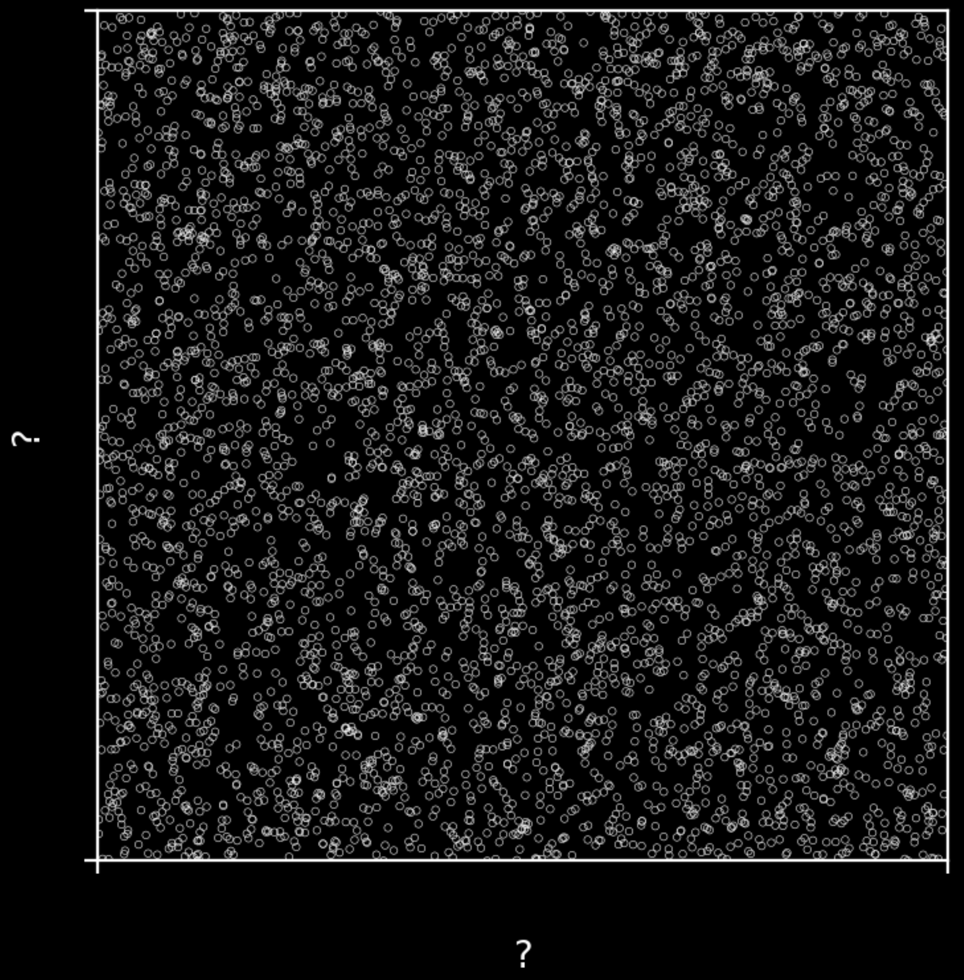
Abundance of capacity

$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T}{\sum_{i=1}^{n} p_i}$$



Setup

(1) Normalised capacity

Abundance of capacity

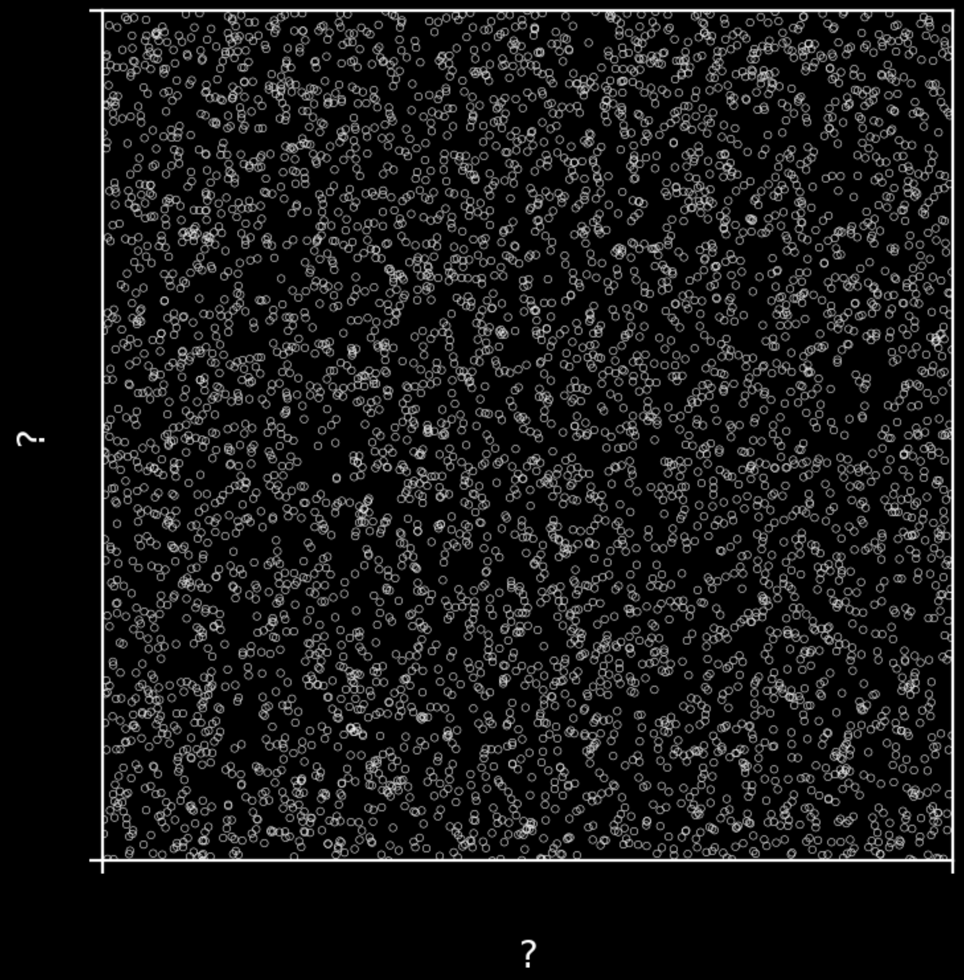
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



Setup

(1) Normalised capacity

Abundance of capacity

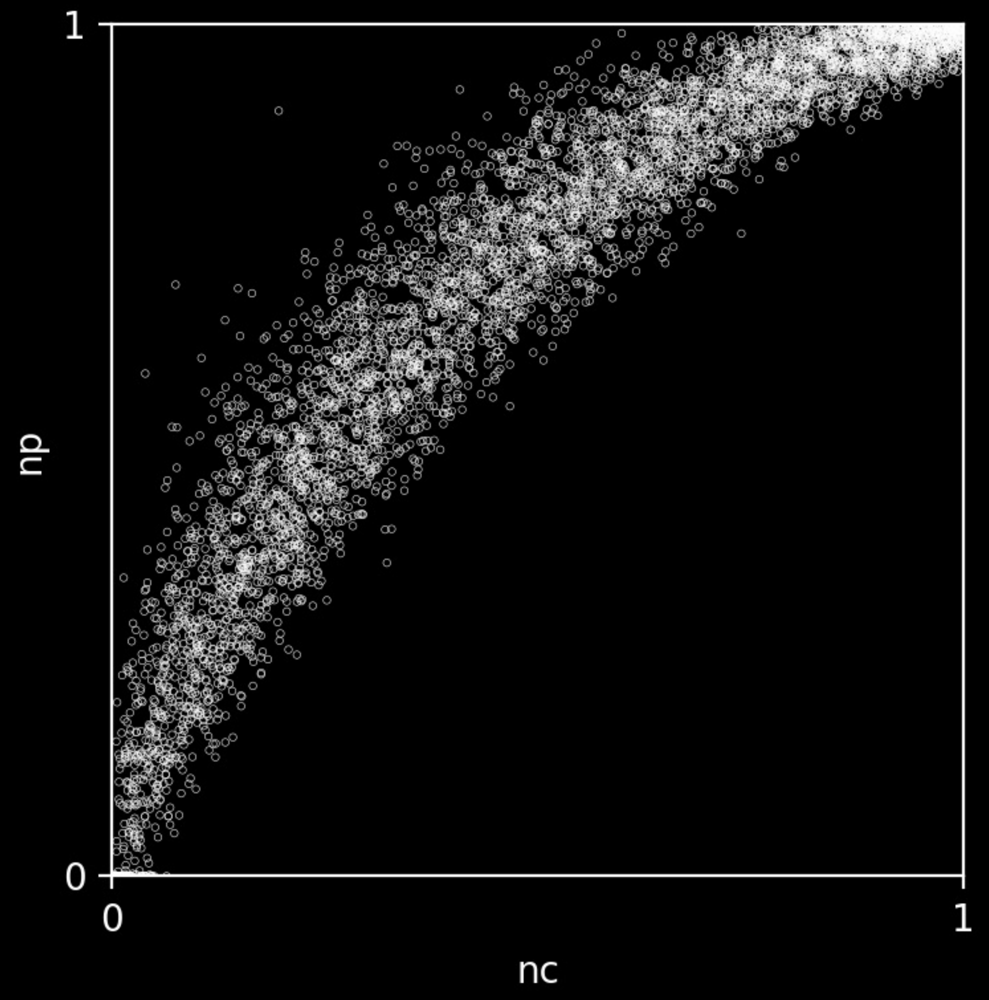
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



Setup

(1) Normalised capacity

Abundance of capacity

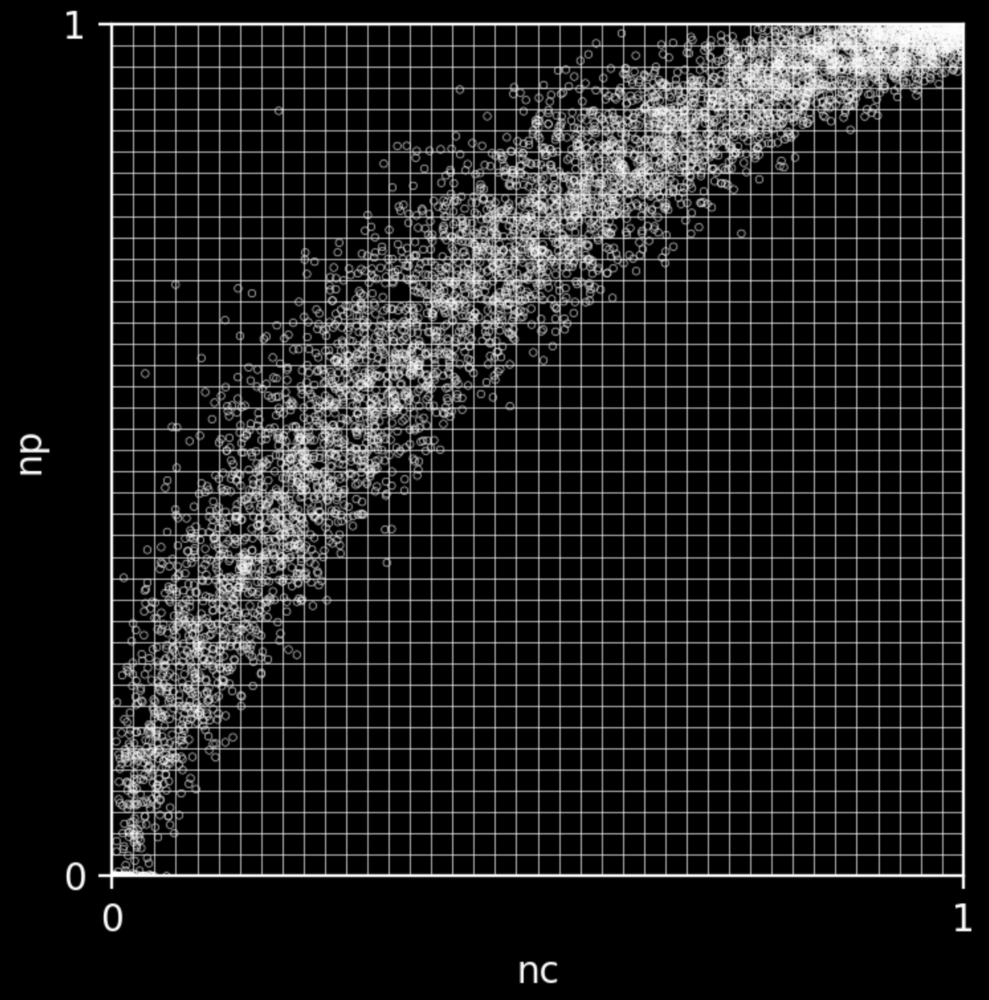
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



Instance distribution

(1) Normalised capacity

Abundance of capacity

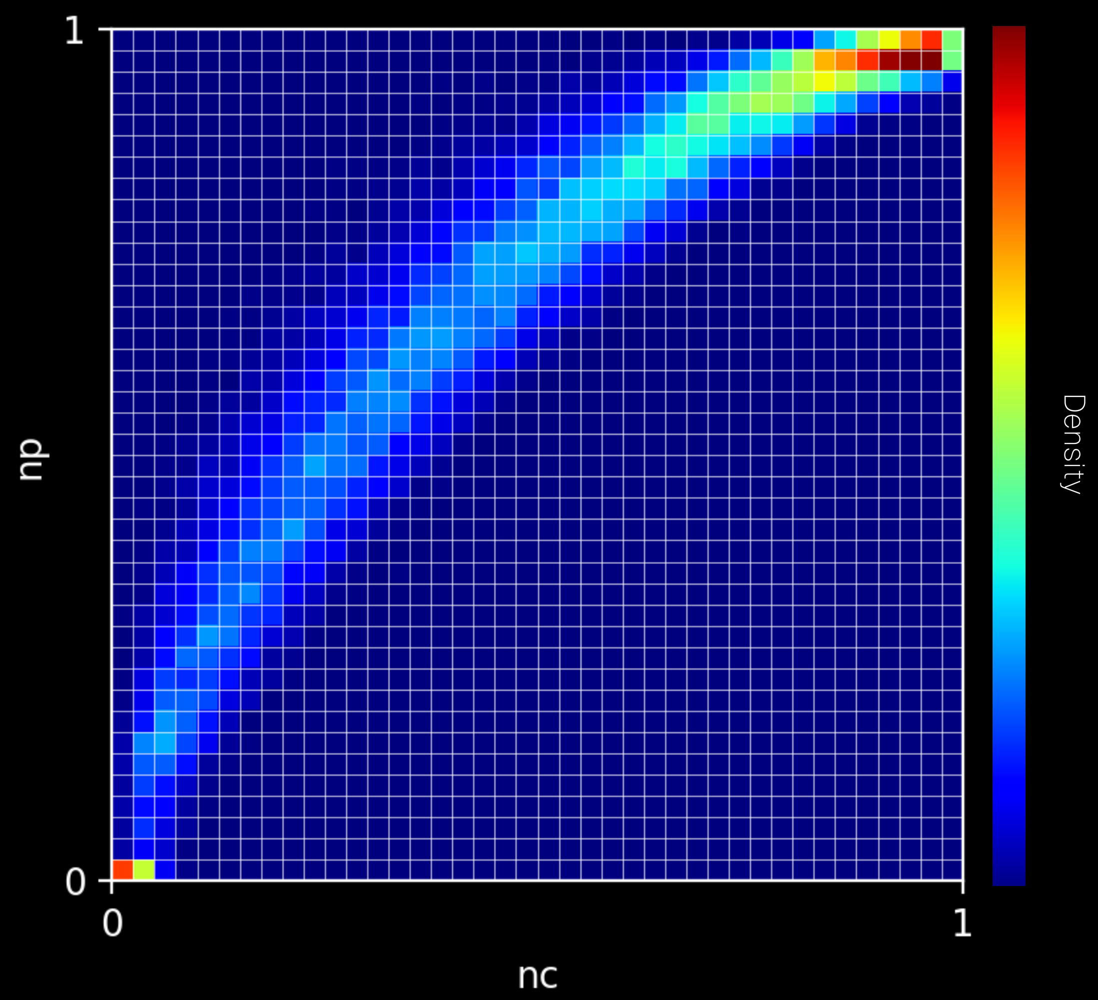
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



Instance distribution

(1) Normalised capacity

Abundance of capacity

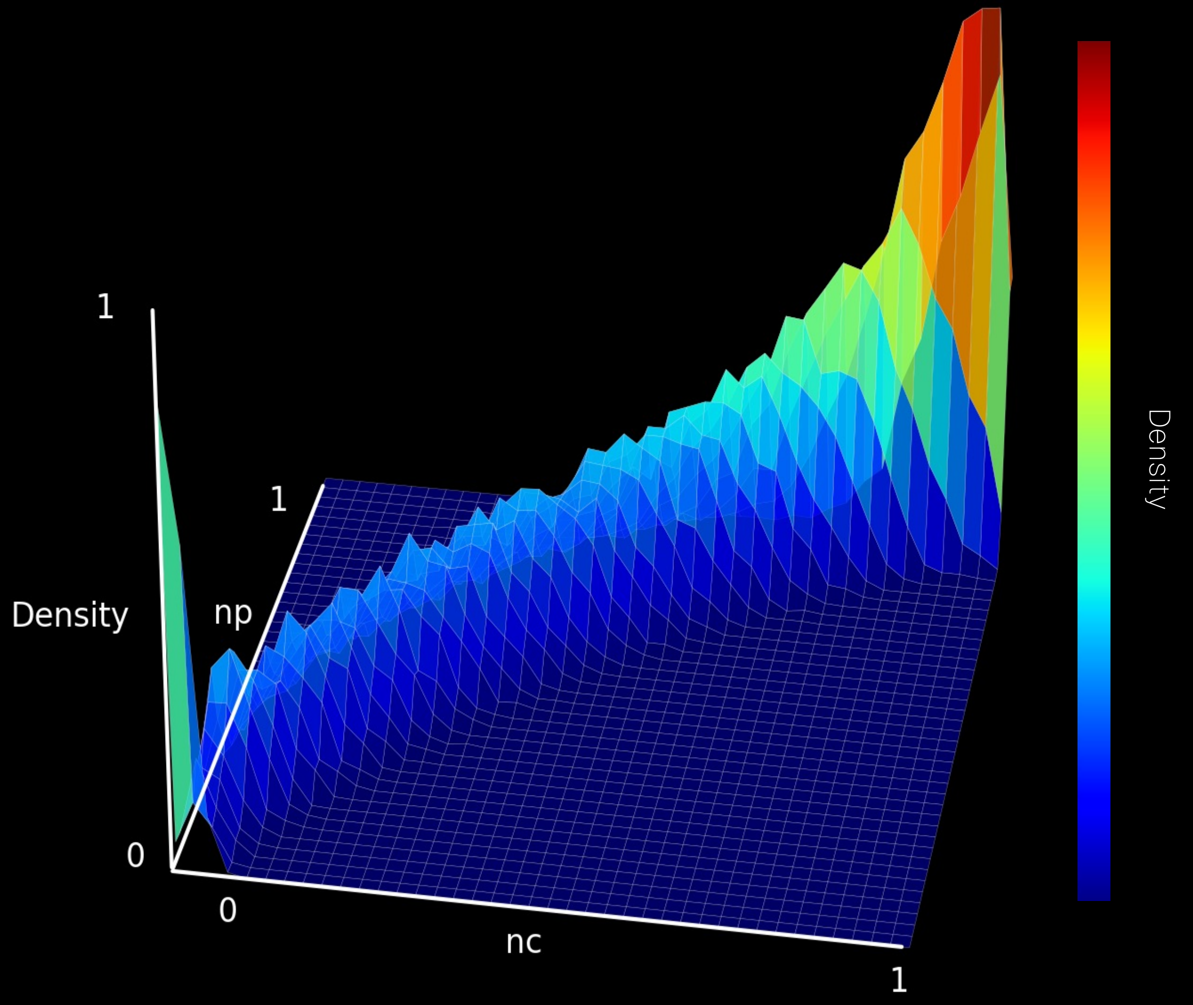
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



Instance distribution

(1) Normalised capacity

Abundance of capacity

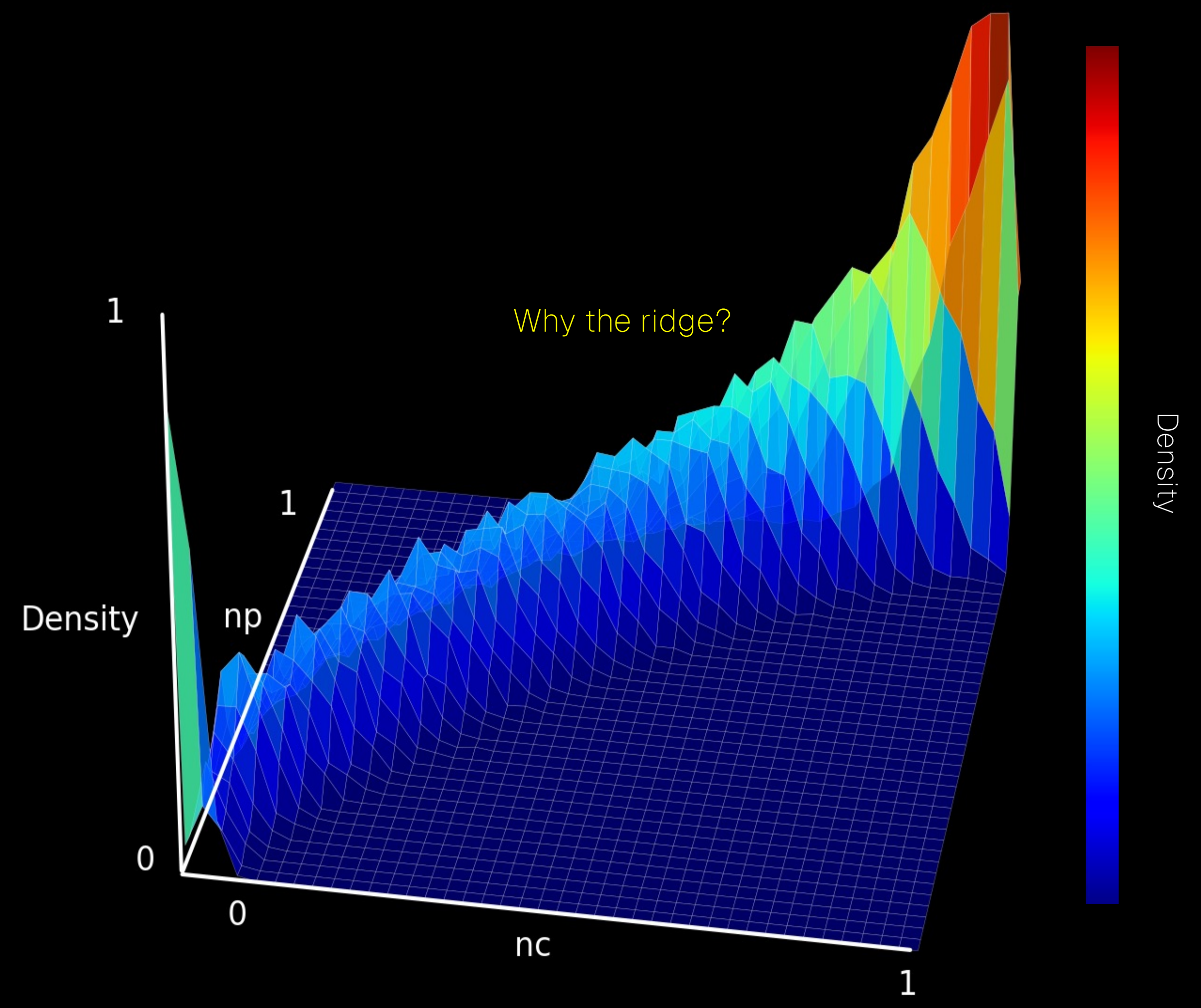
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



Instance distribution

(1) Normalised capacity

Abundance of capacity

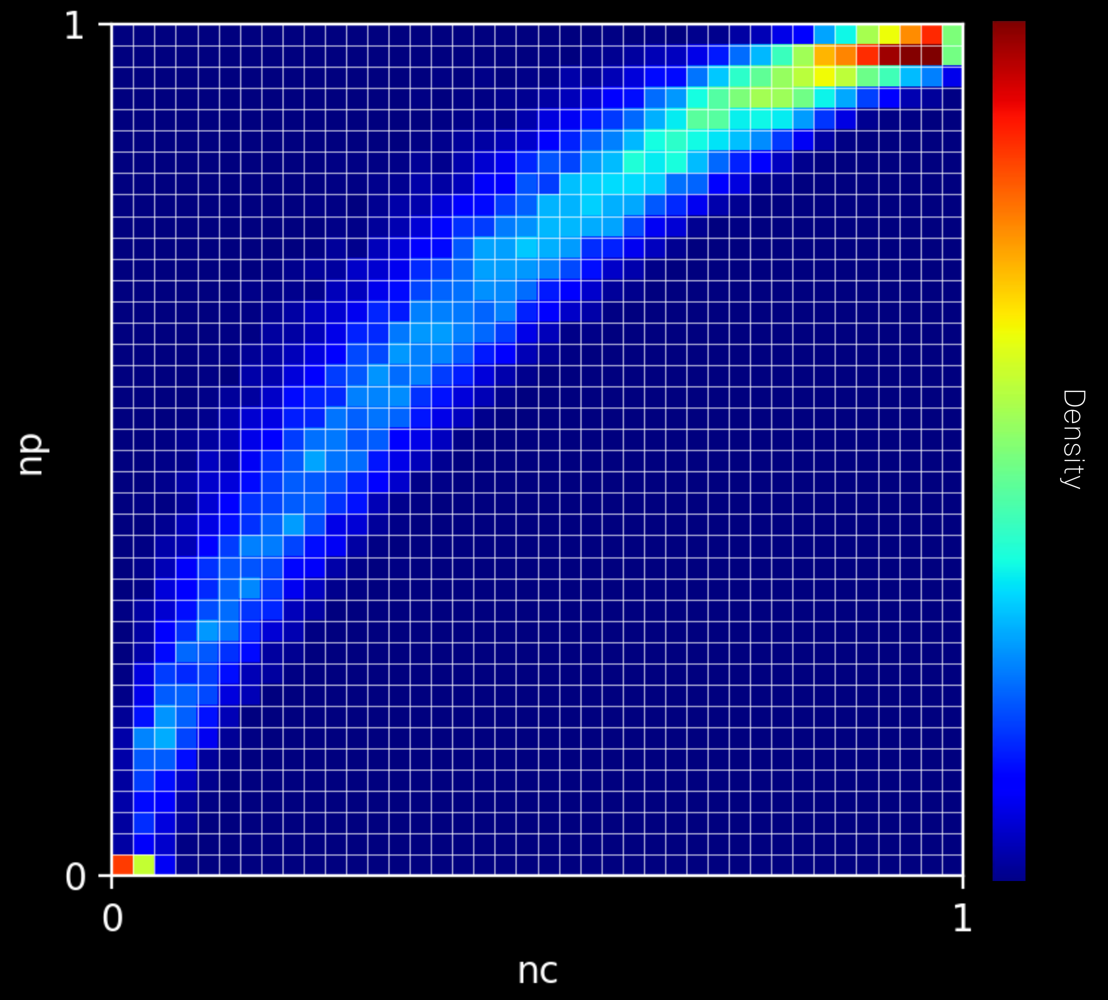
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



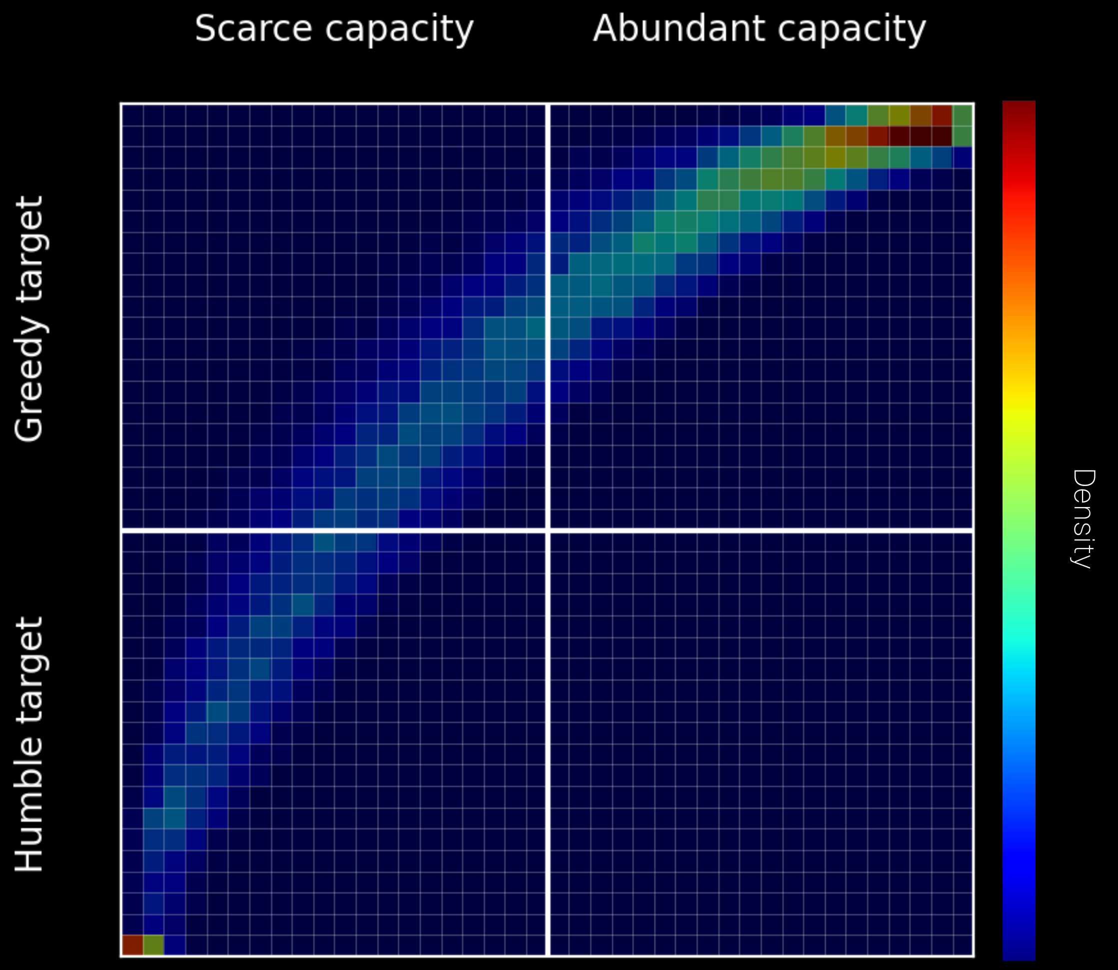
Instance distribution

Available items

?

Optimal knapsack

?



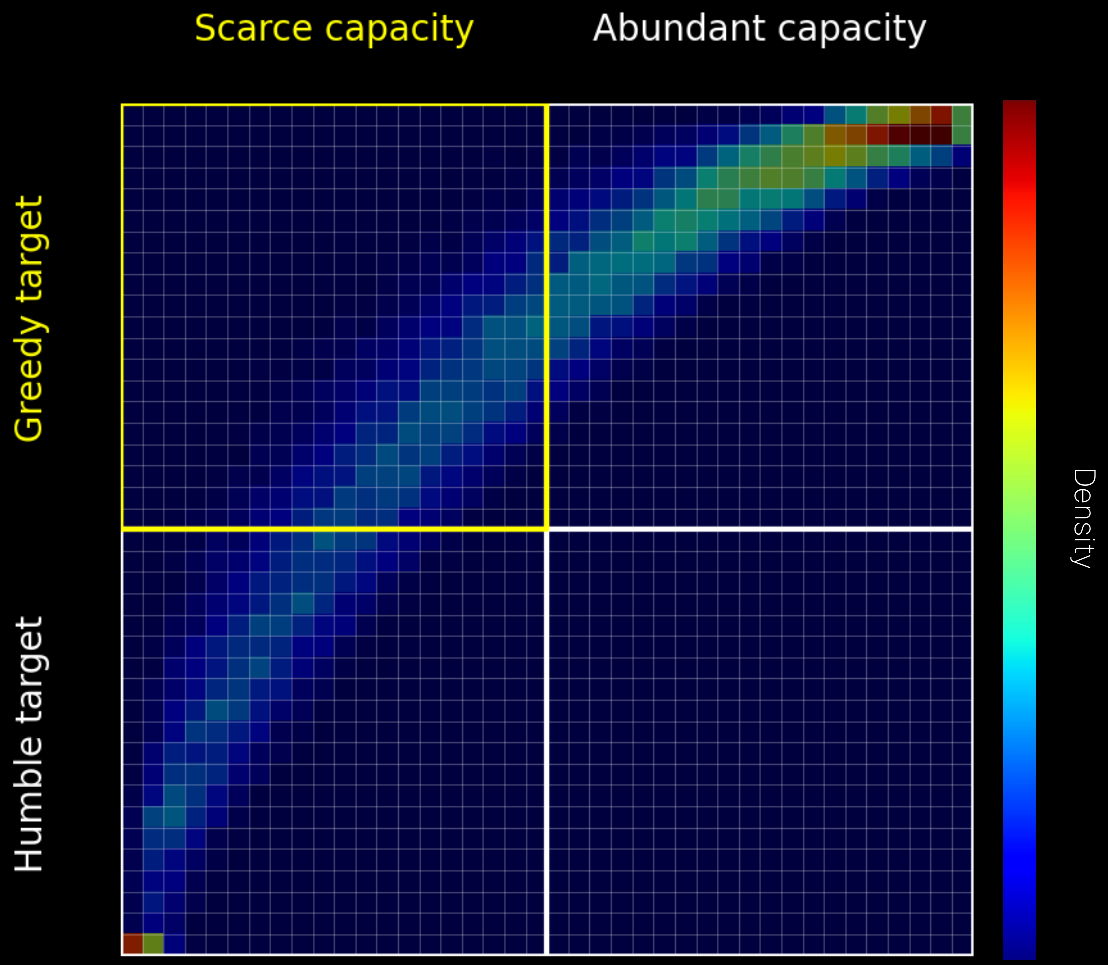
Instance distribution

Available items

?

Optimal knapsack

?

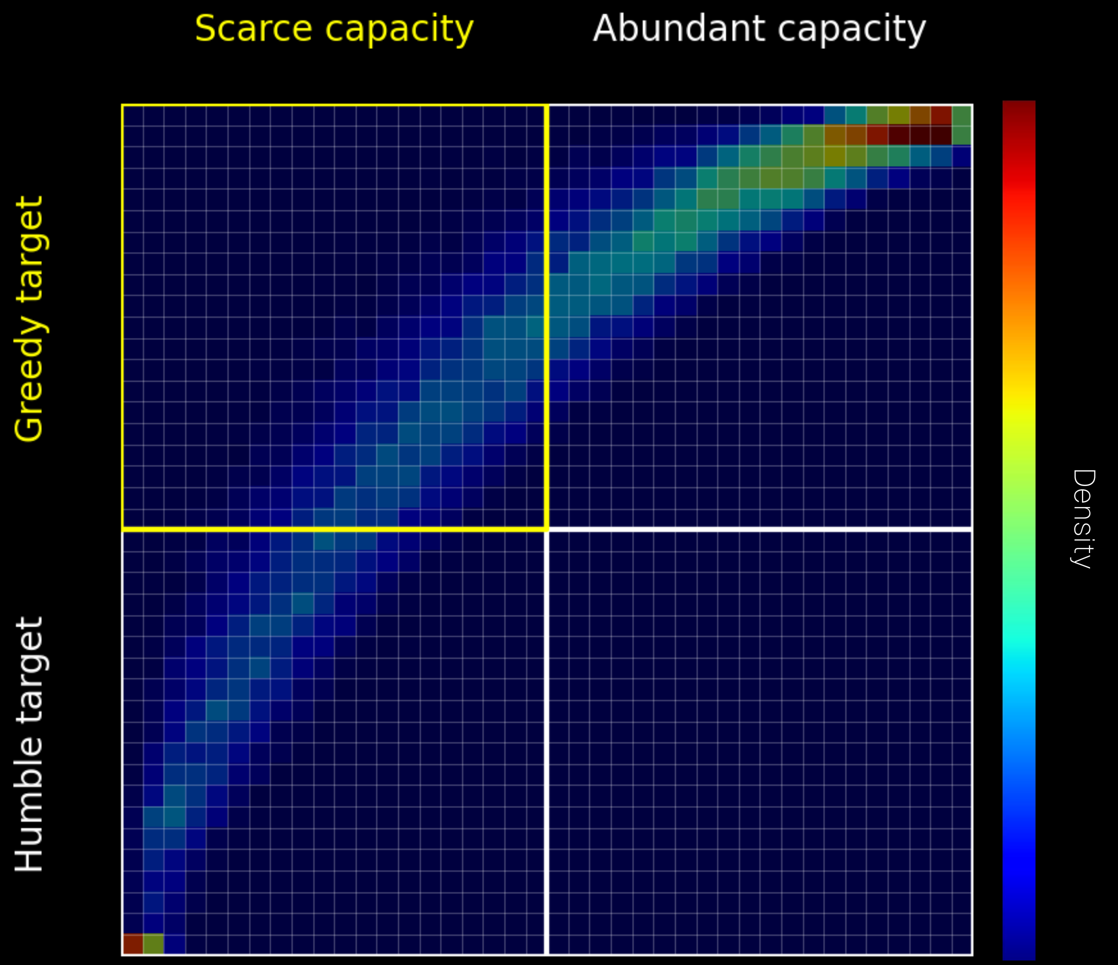


Instance distribution

Available items

Optimal knapsack

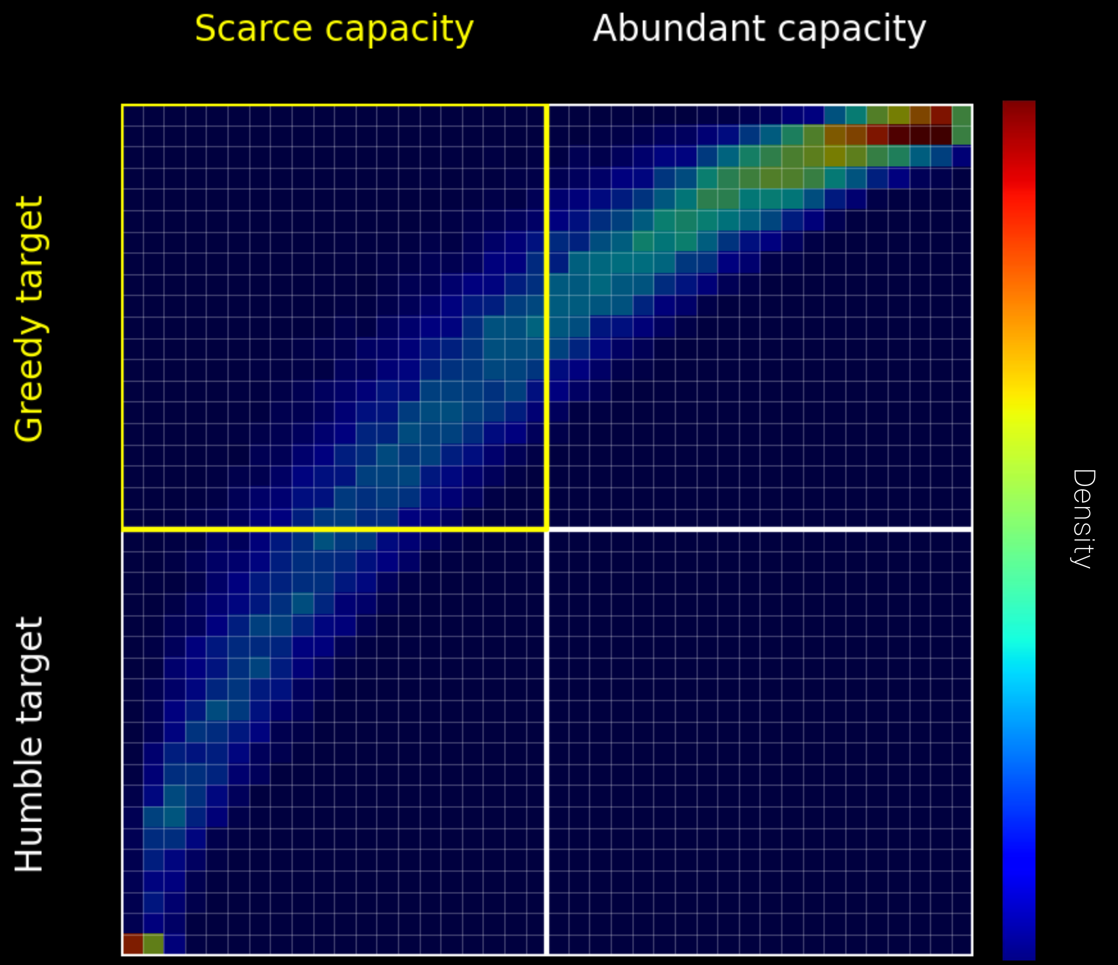
?



Instance distribution

Available items

Optimal knapsack



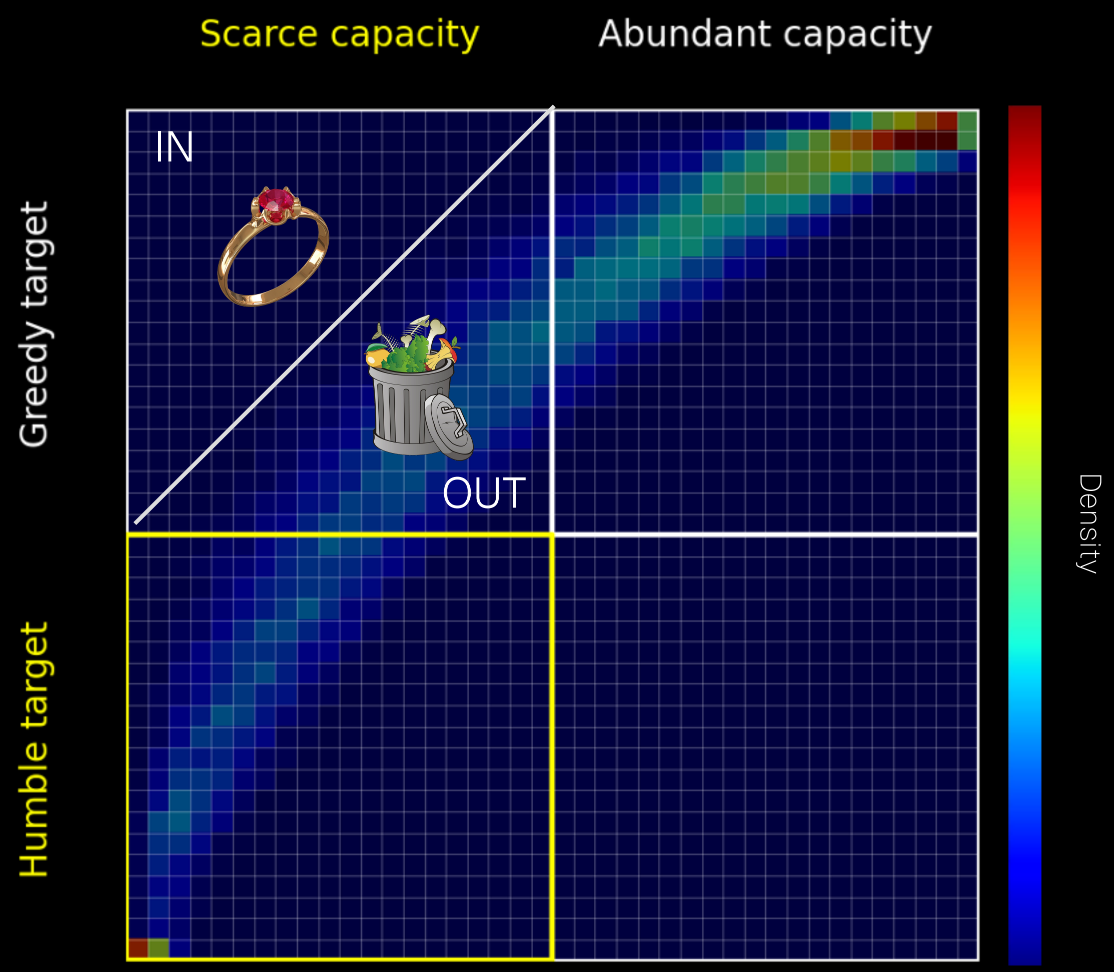
Instance distribution

Available items

?

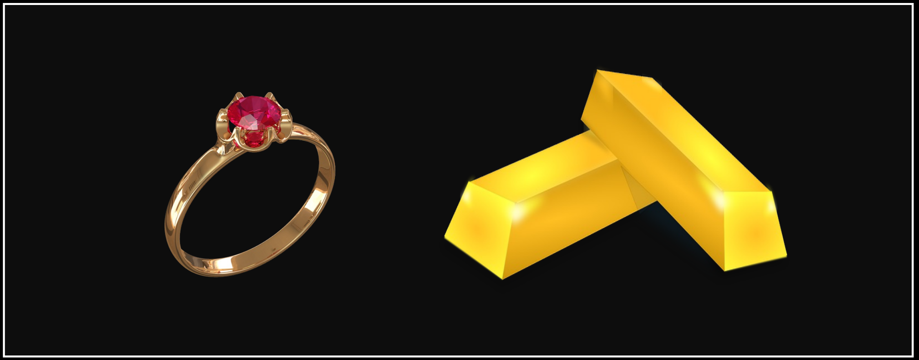
Optimal knapsack

?



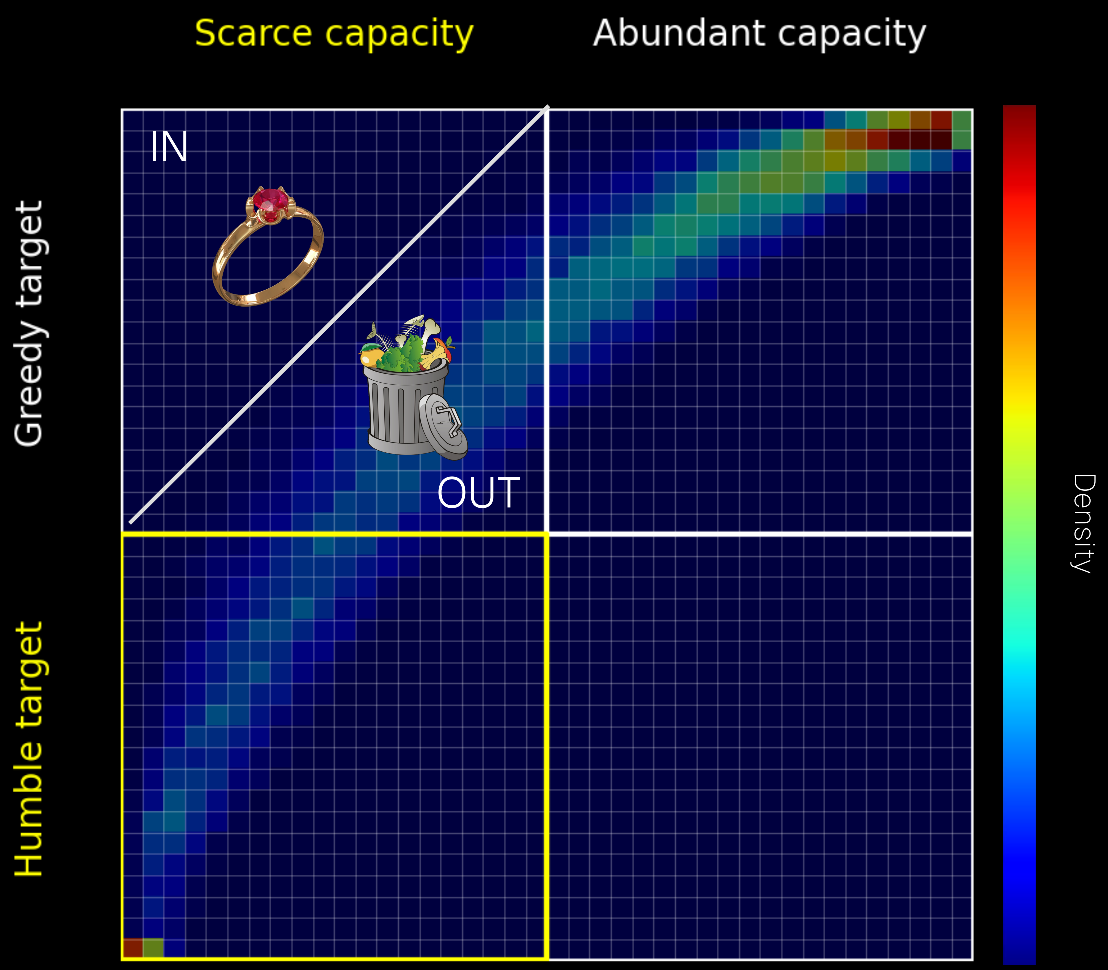
Instance distribution

Available items



Optimal knapsack

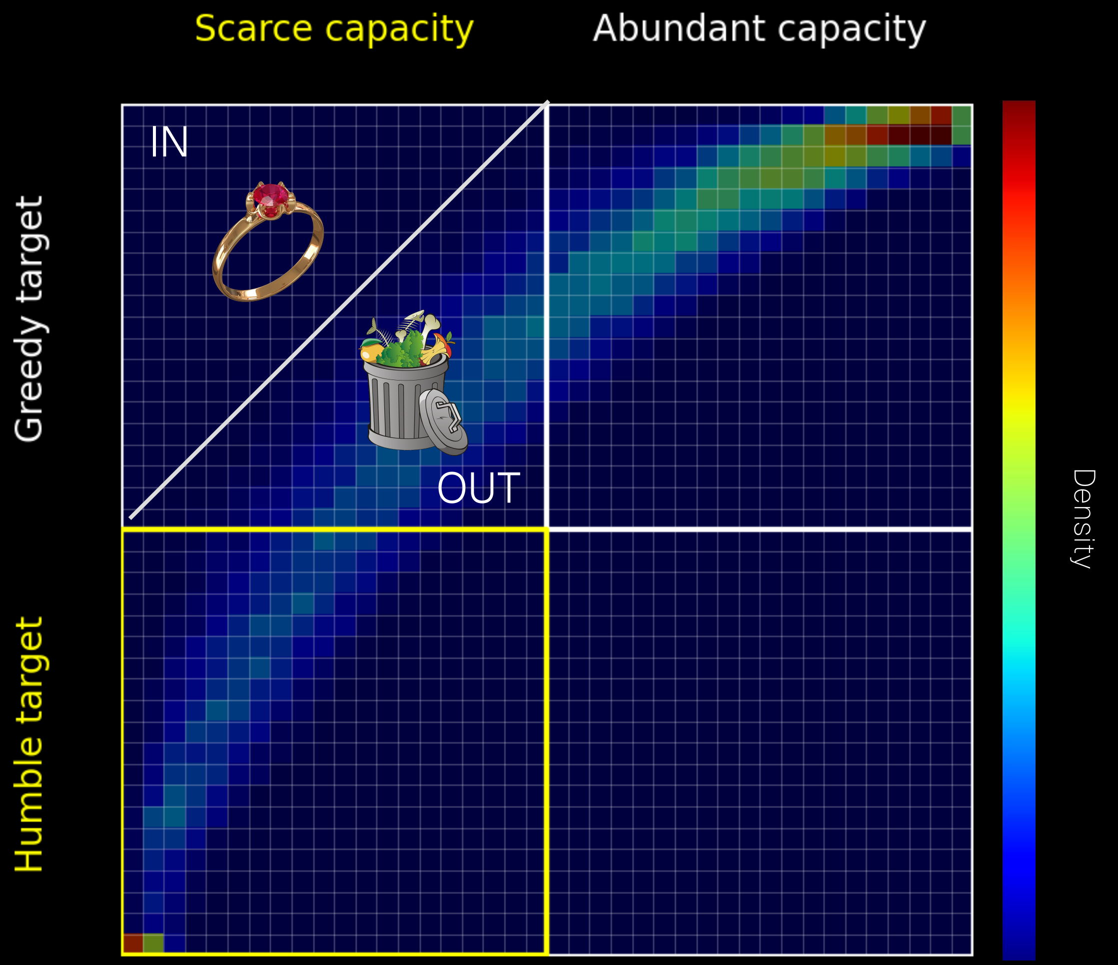
?



Instance distribution

Available items

Optimal knapsack



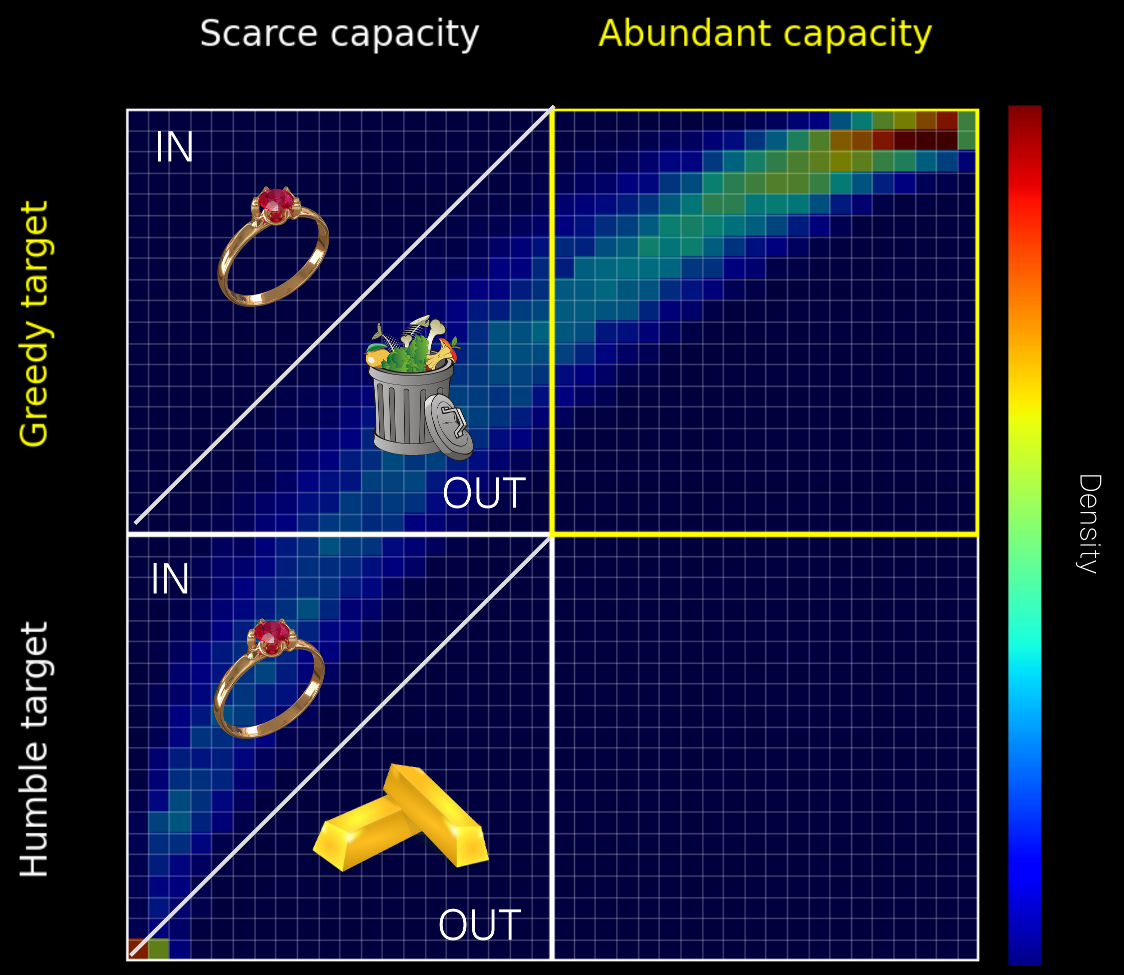
Instance distribution

Available items

?

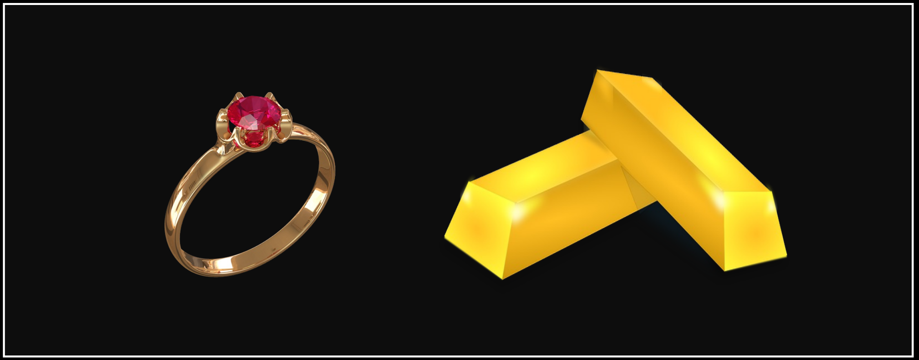
Optimal knapsack

?



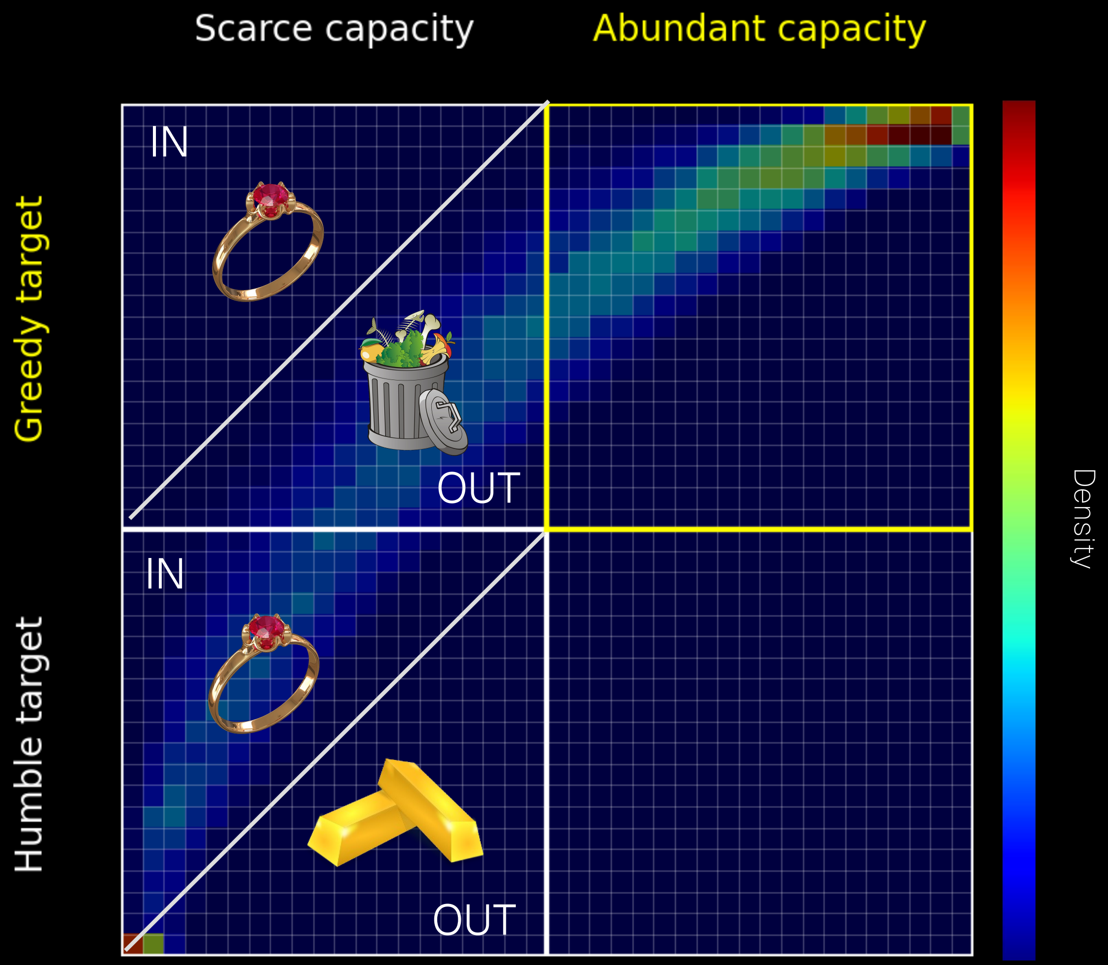
Instance distribution

Available items



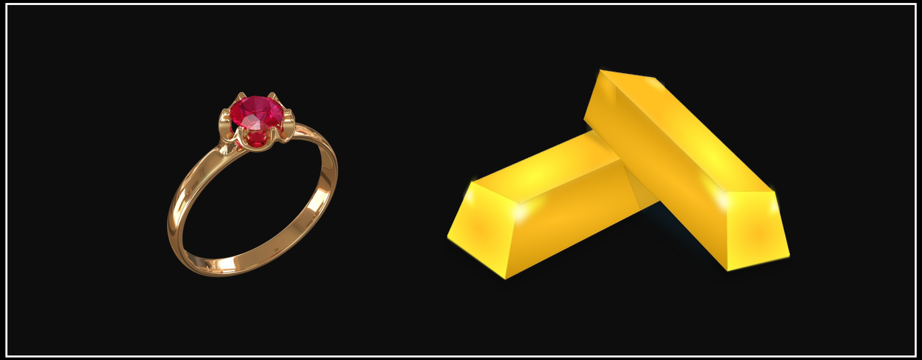
Optimal knapsack

?

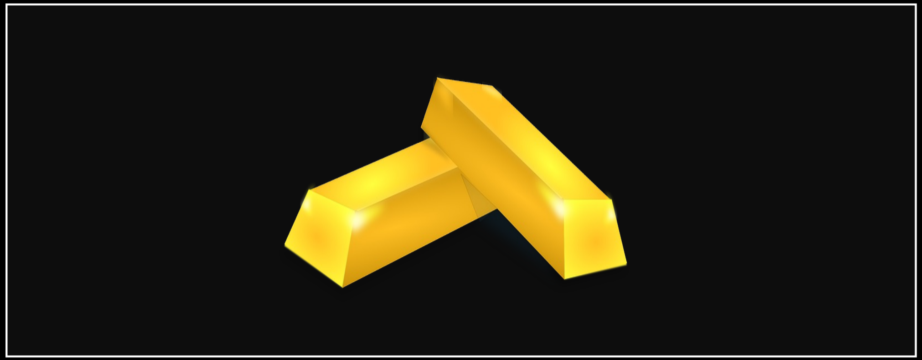


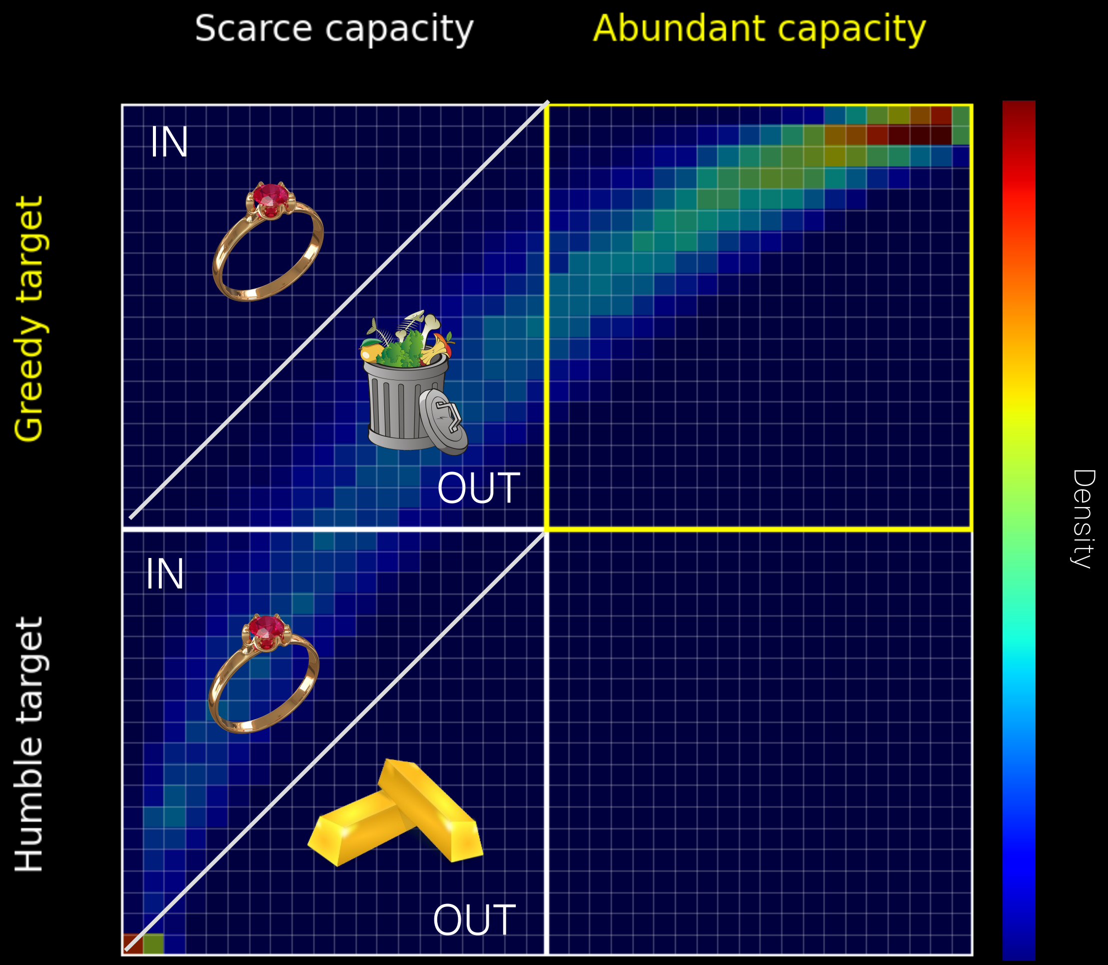
Instance distribution

Available items



Optimal knapsack





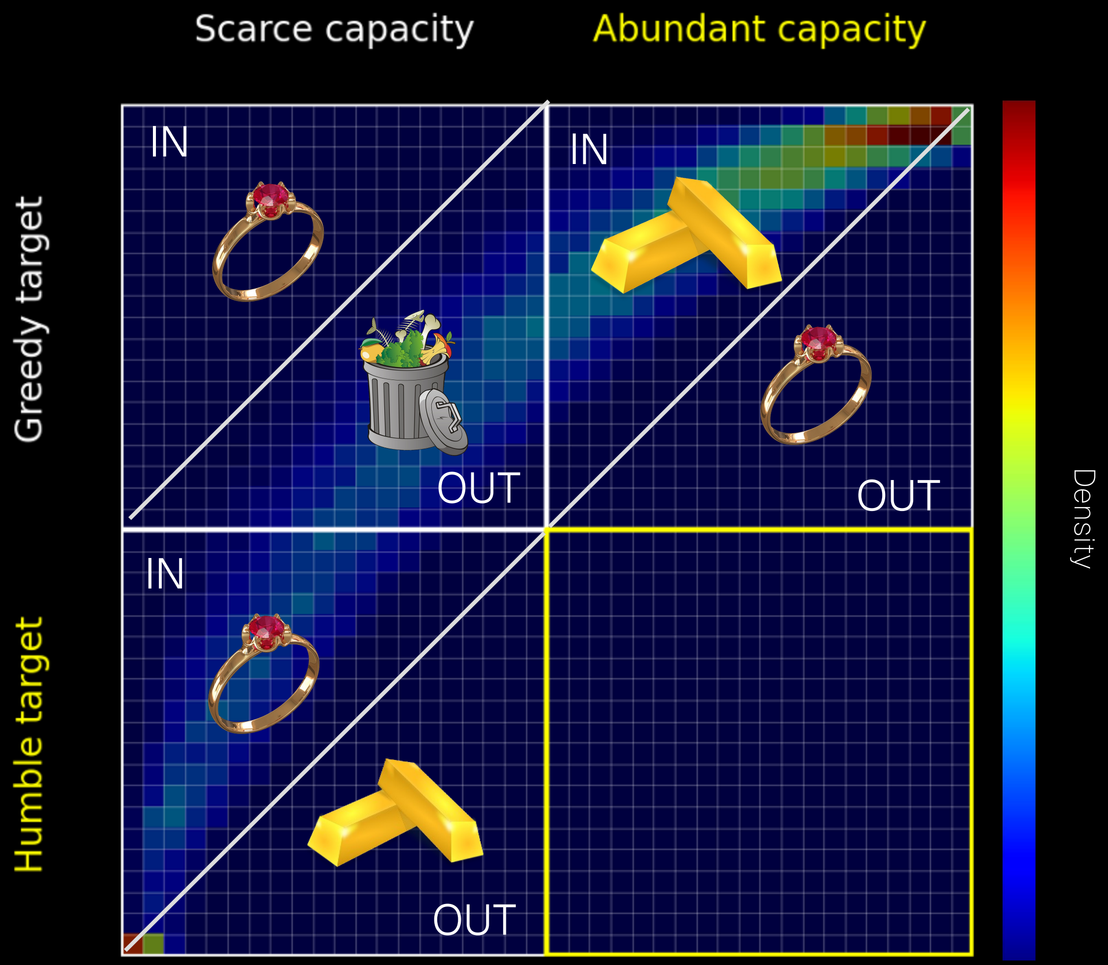
Instance distribution

Available items

?

Optimal knapsack

?

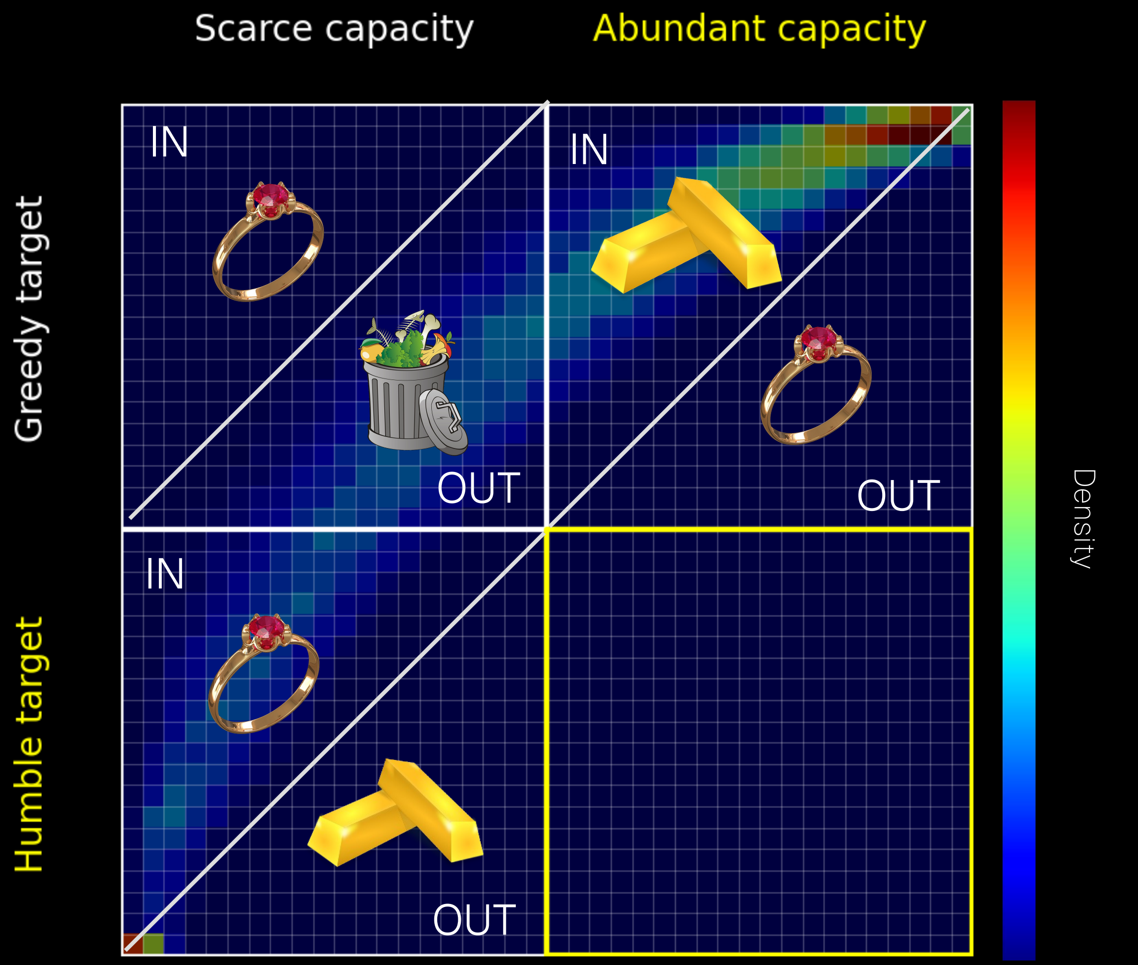


Instance distribution

Available items

Optimal knapsack

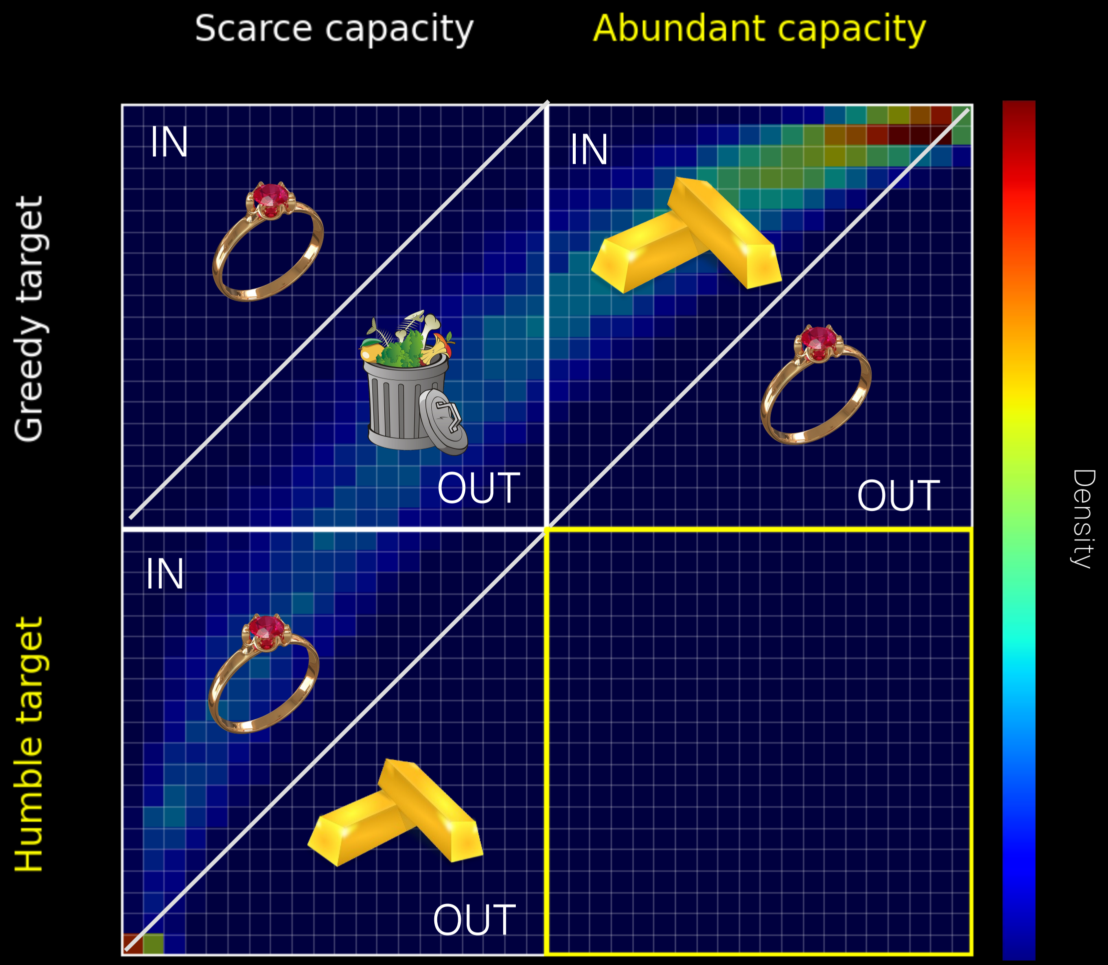
?



Instance distribution

Available items

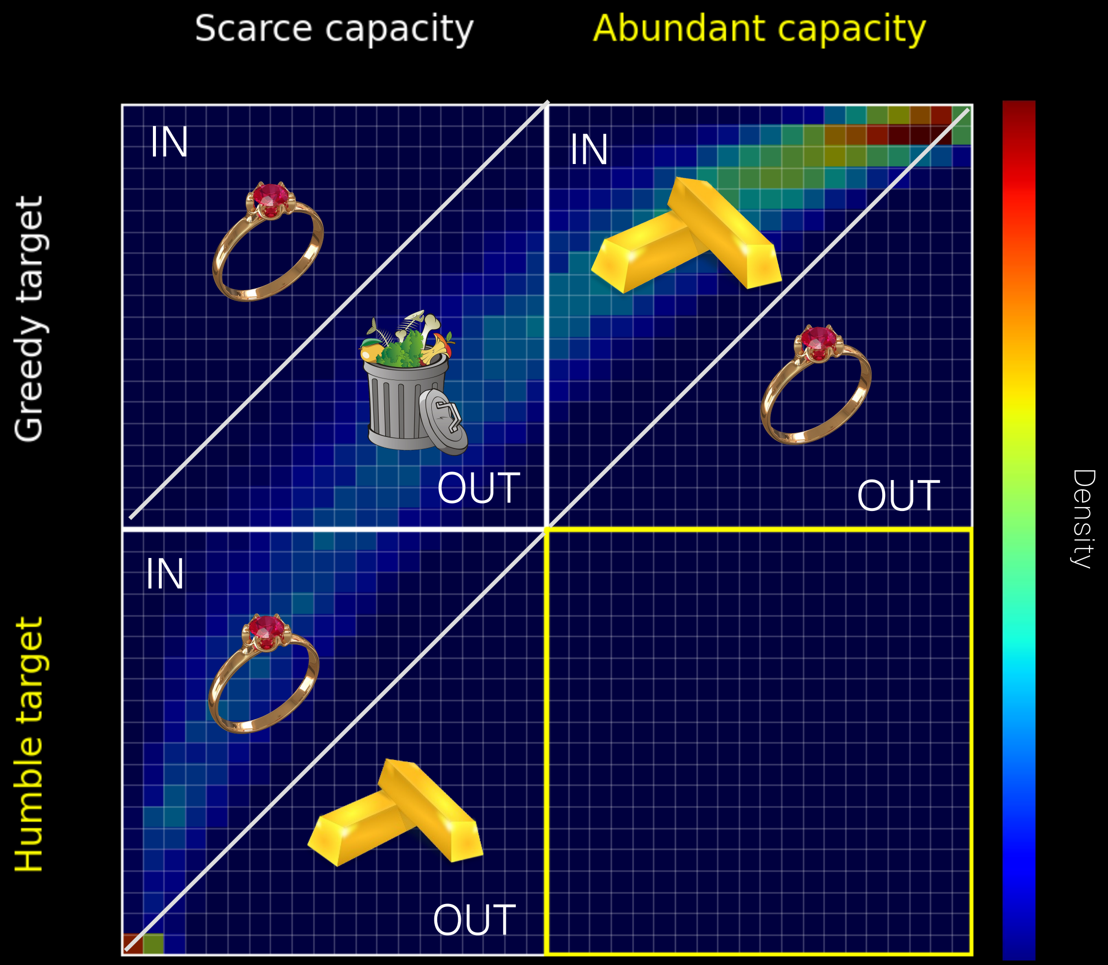
Optimal knapsack



Instance distribution

Available items

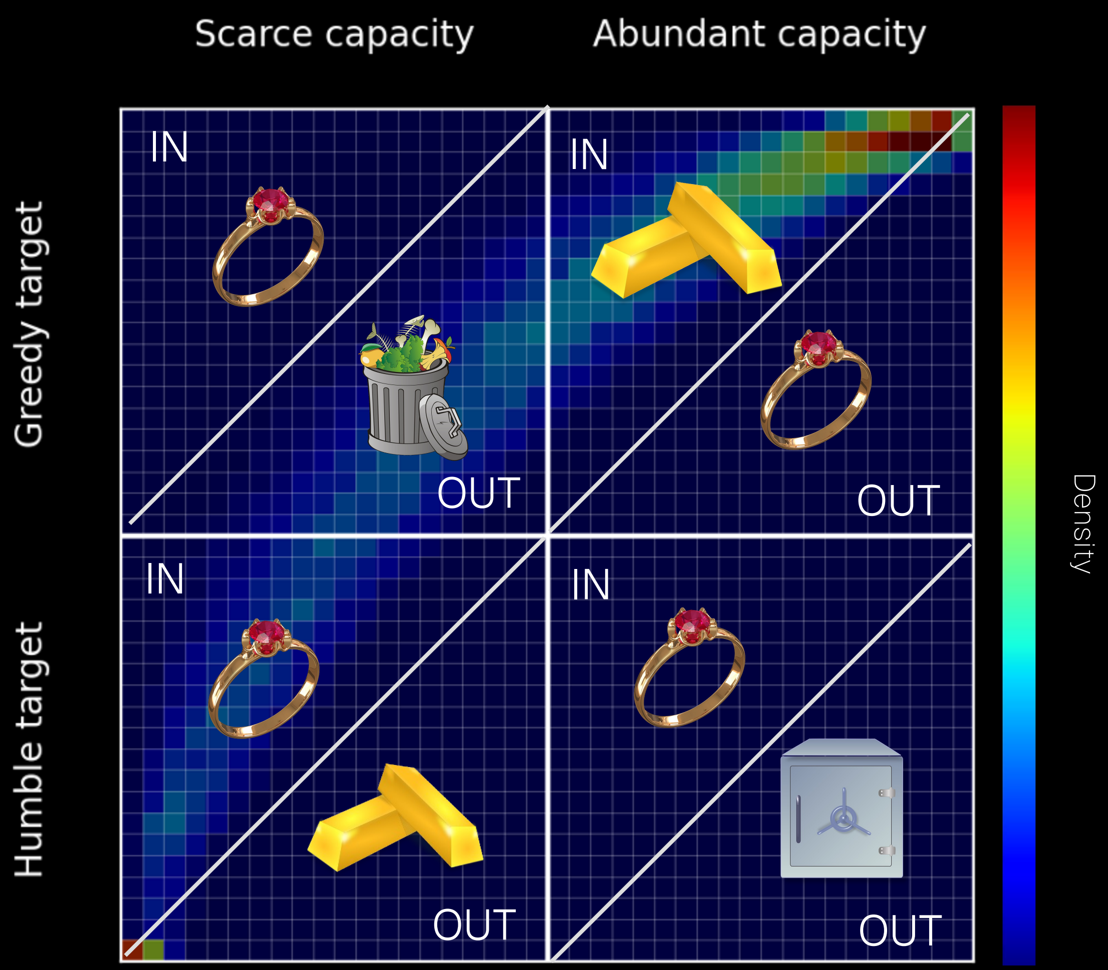
Optimal knapsack



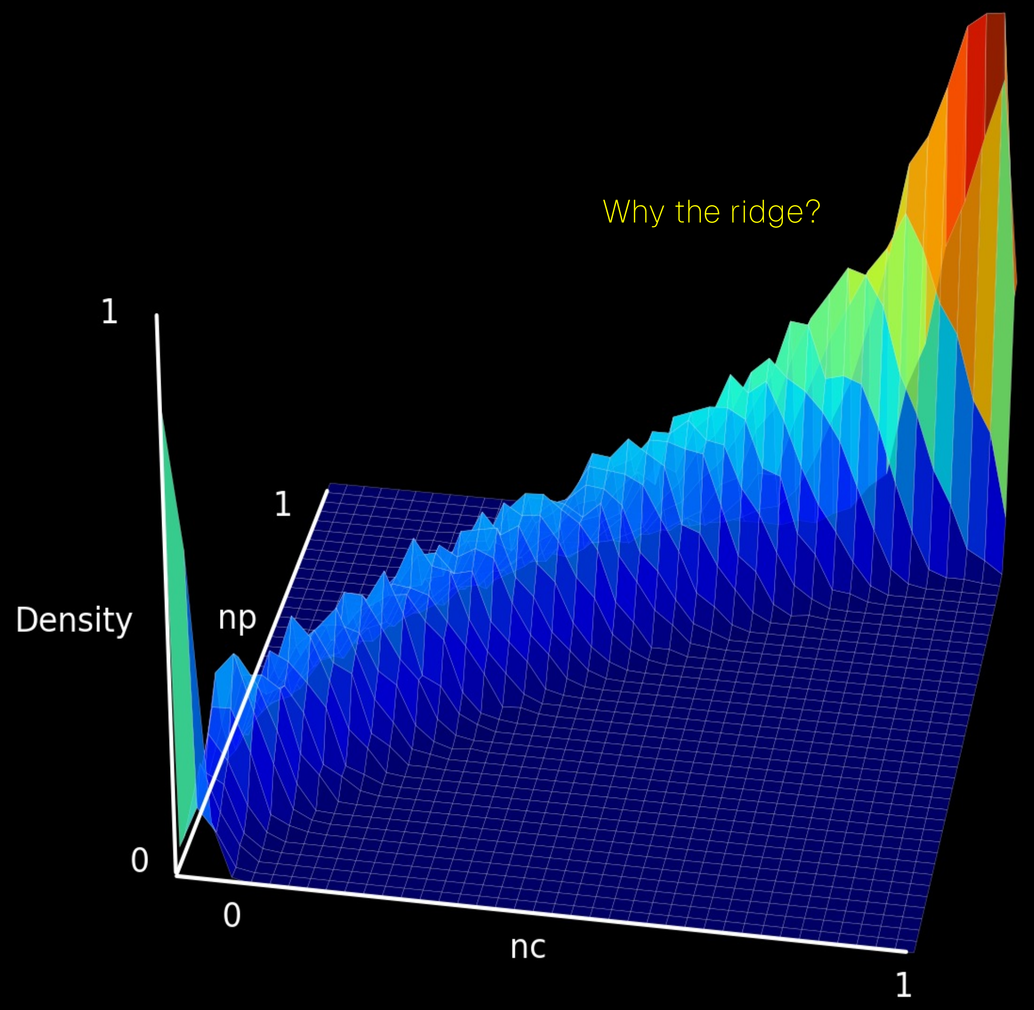
Instance distribution

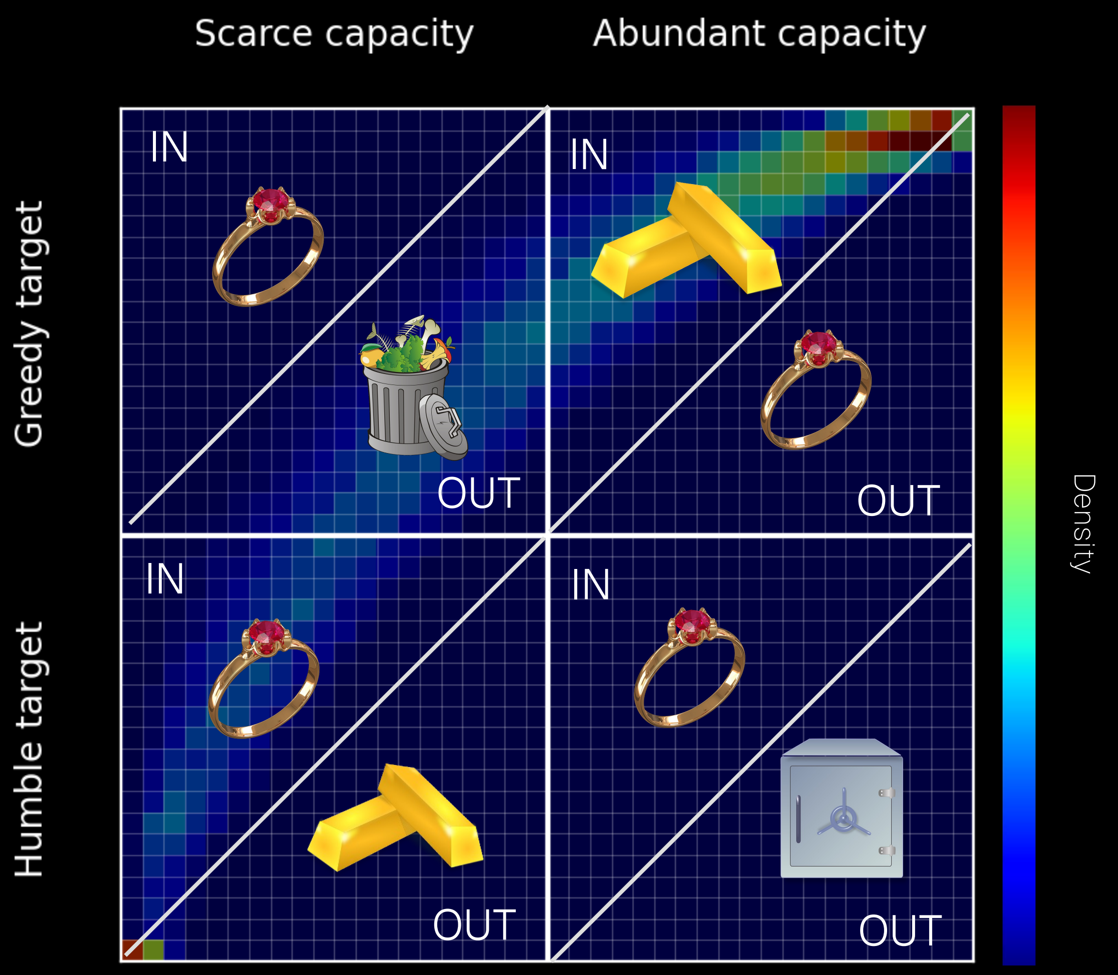
Available items

Optimal knapsack

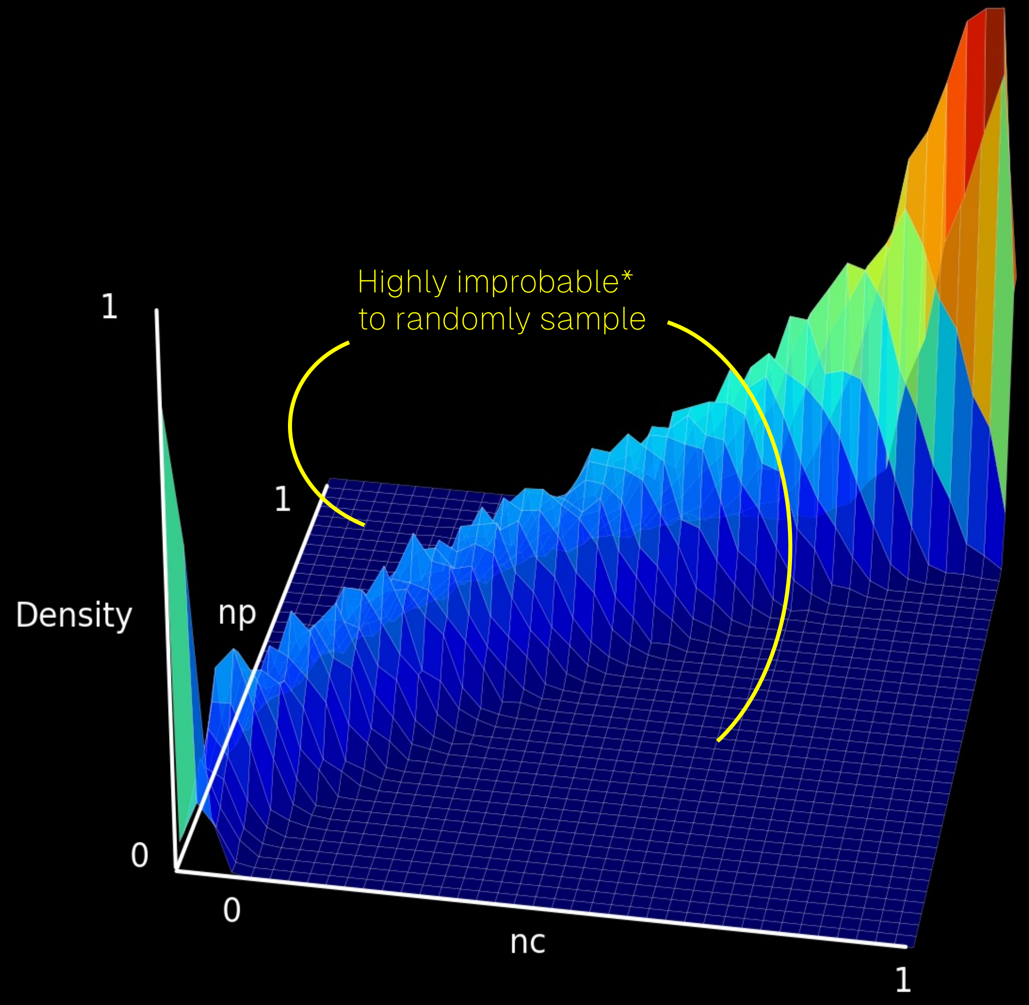


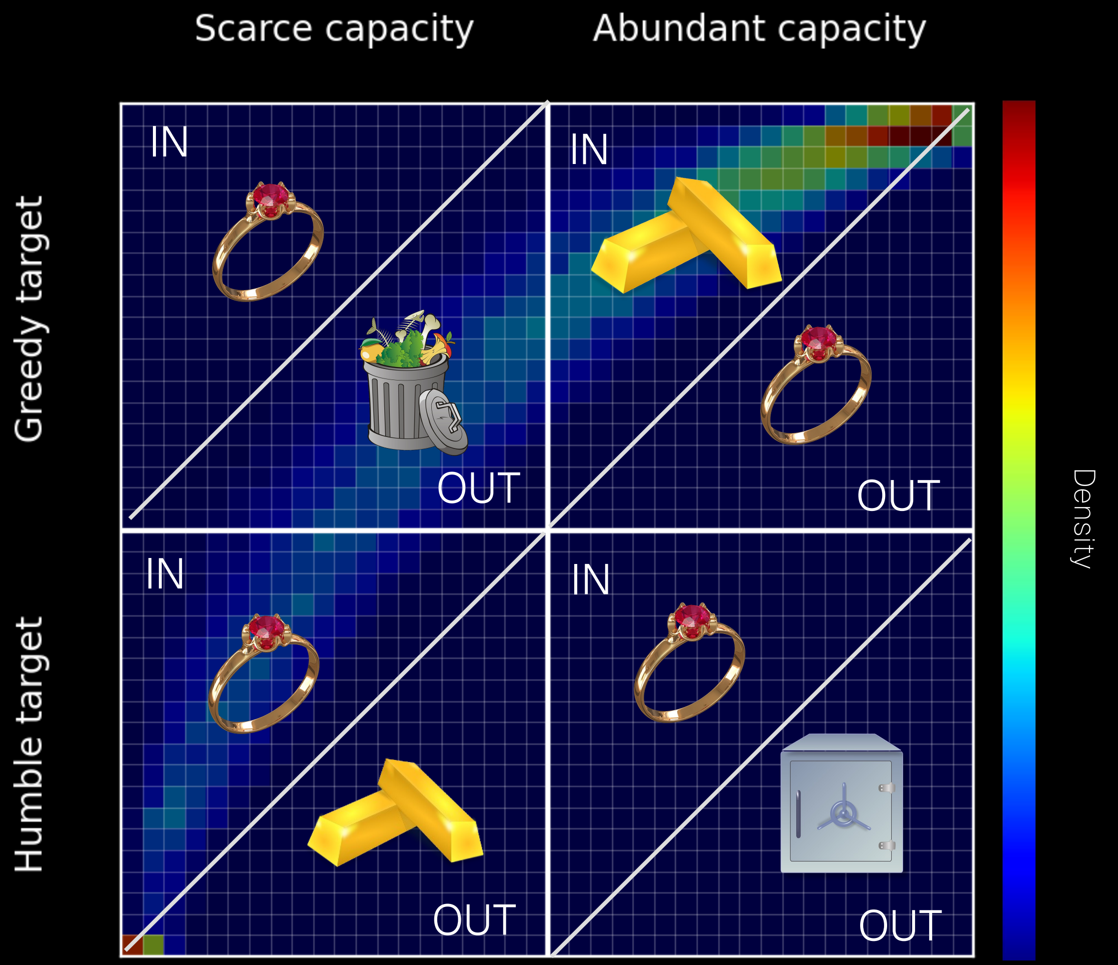
Instance distribution





Instance distribution





What can we learn about the decision phase transition?

(1) Normalised capacity

Abundance of capacity

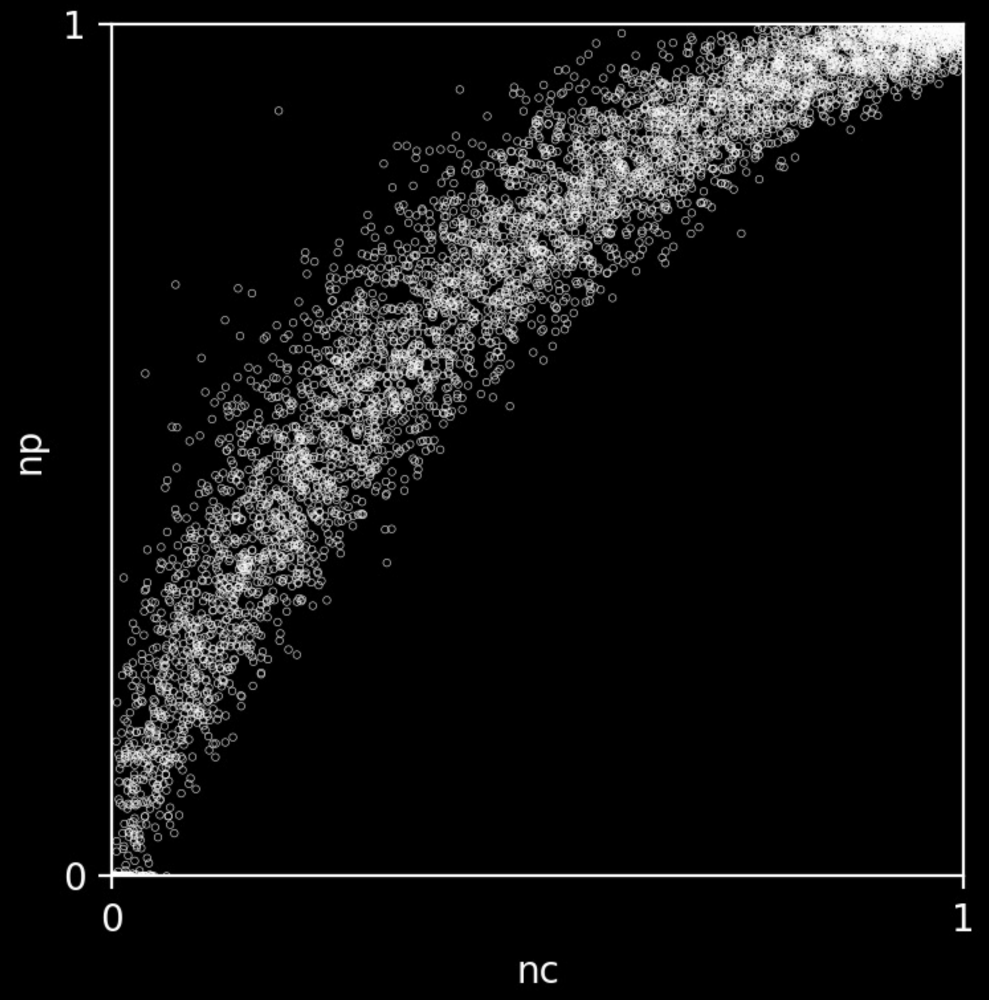
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



Phase transition ≈ optimisation instances

(1) Normalised capacity

Abundance of capacity

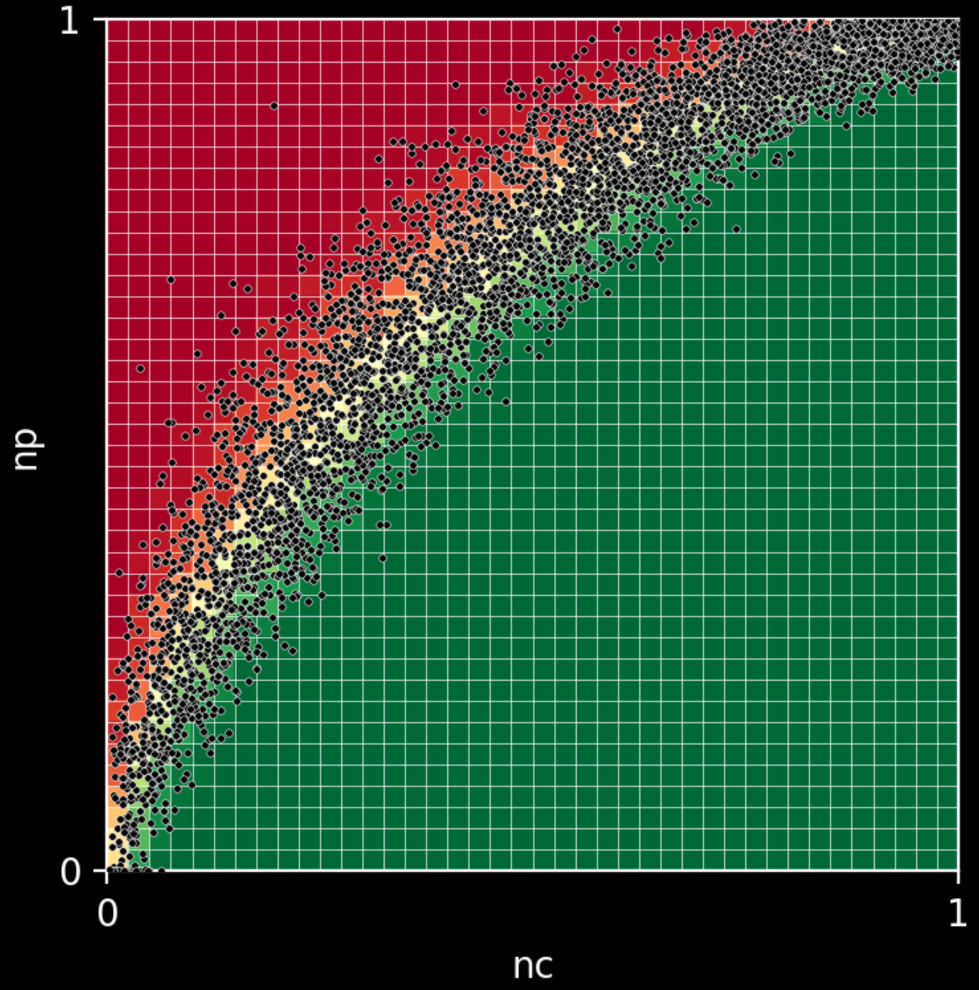
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value

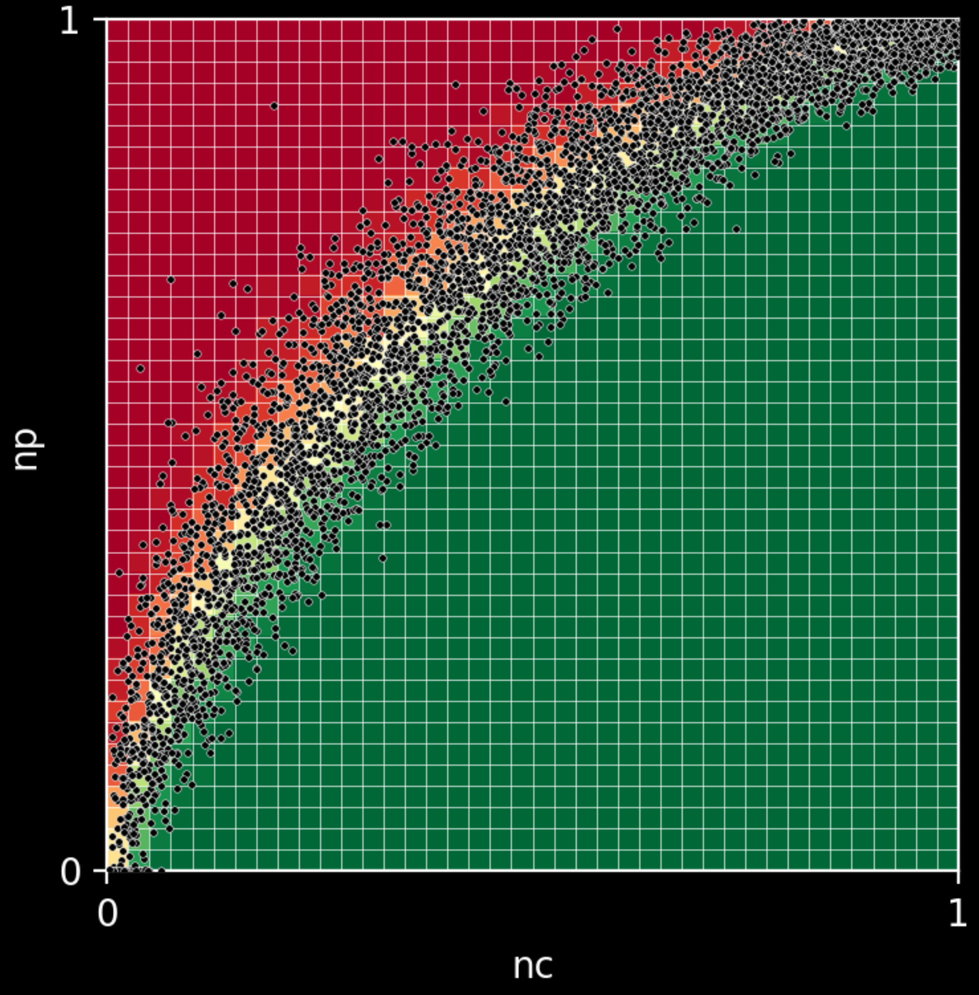


Phase transition ≈ optimisation instances

- Decision problems in the phase transition are pseudo-optimisation problems.

- Formally, a decision problem $\langle C, W, V, T \rangle$ with optimal value v^* is in the phase transition if $\mathrm{T} \approx v^*$.

- Sampling from the phase transition is analogous to sampling pseudo-optimisation problems.



Where are the *really* hard problems?

(1) Normalised capacity

Abundance of capacity

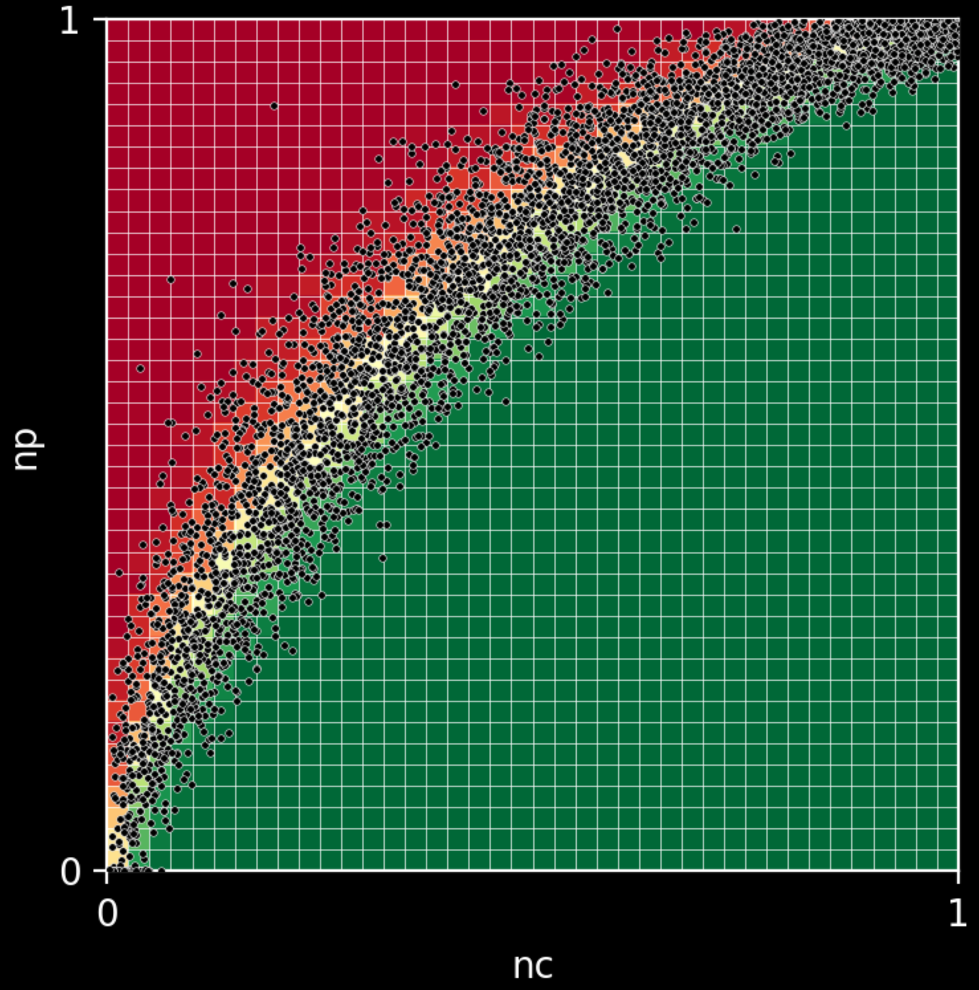
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

Where T^* = optimal value



Where are the *really* hard problems?

(1) Normalised capacity

Abundance of capacity

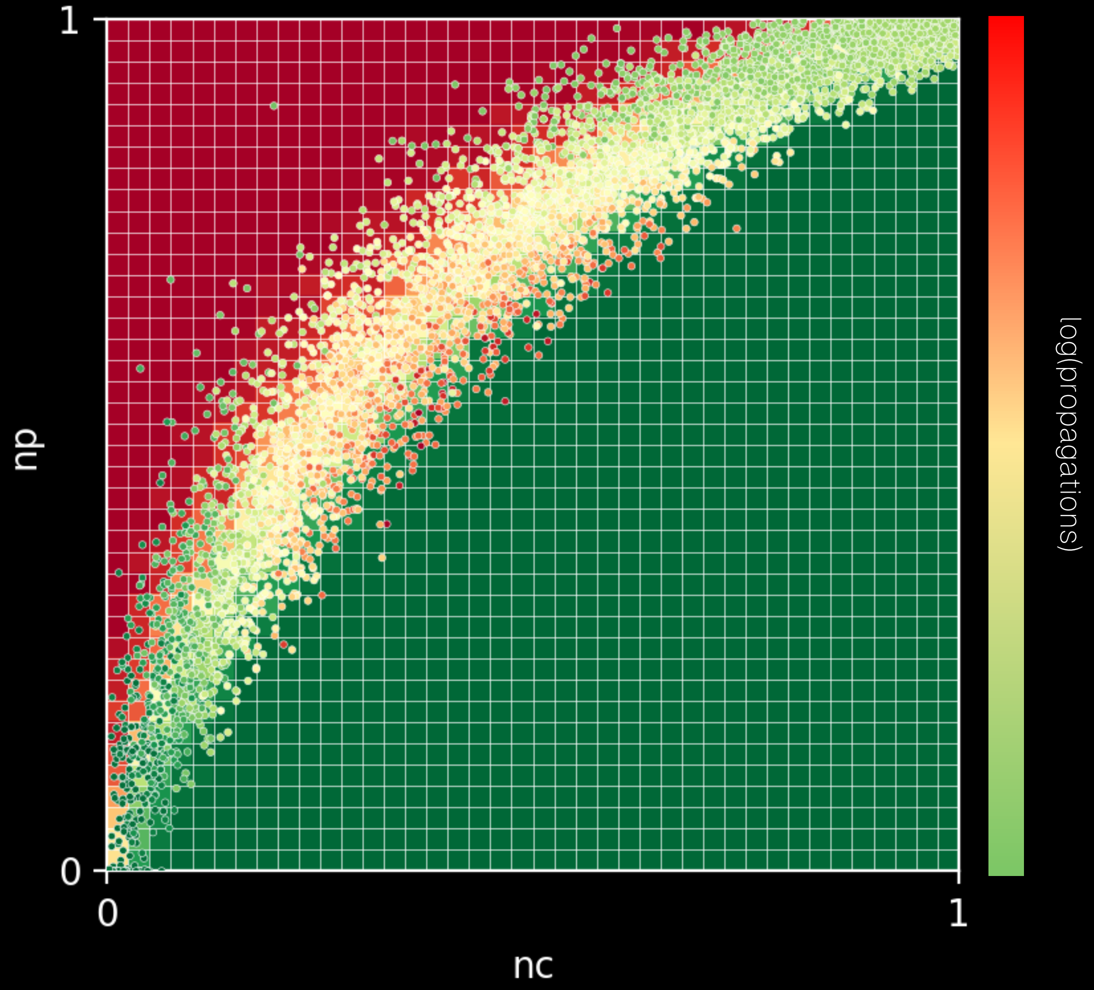
$$nc = \frac{C}{\sum_{i=1}^{n} w_i}$$

(2) Normalised profit

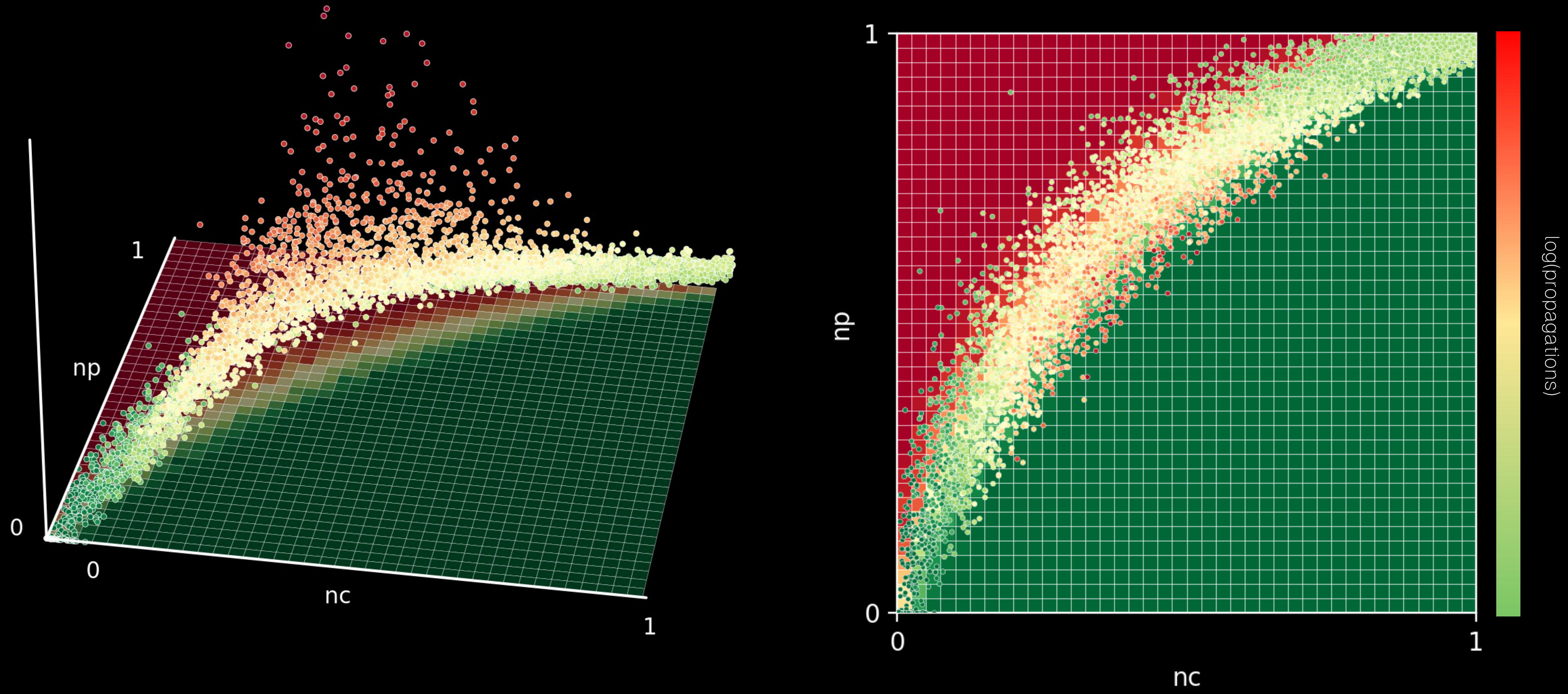
Greediness of target profit

$$np = \frac{T^*}{\sum_{i=1}^{n} p_i}$$

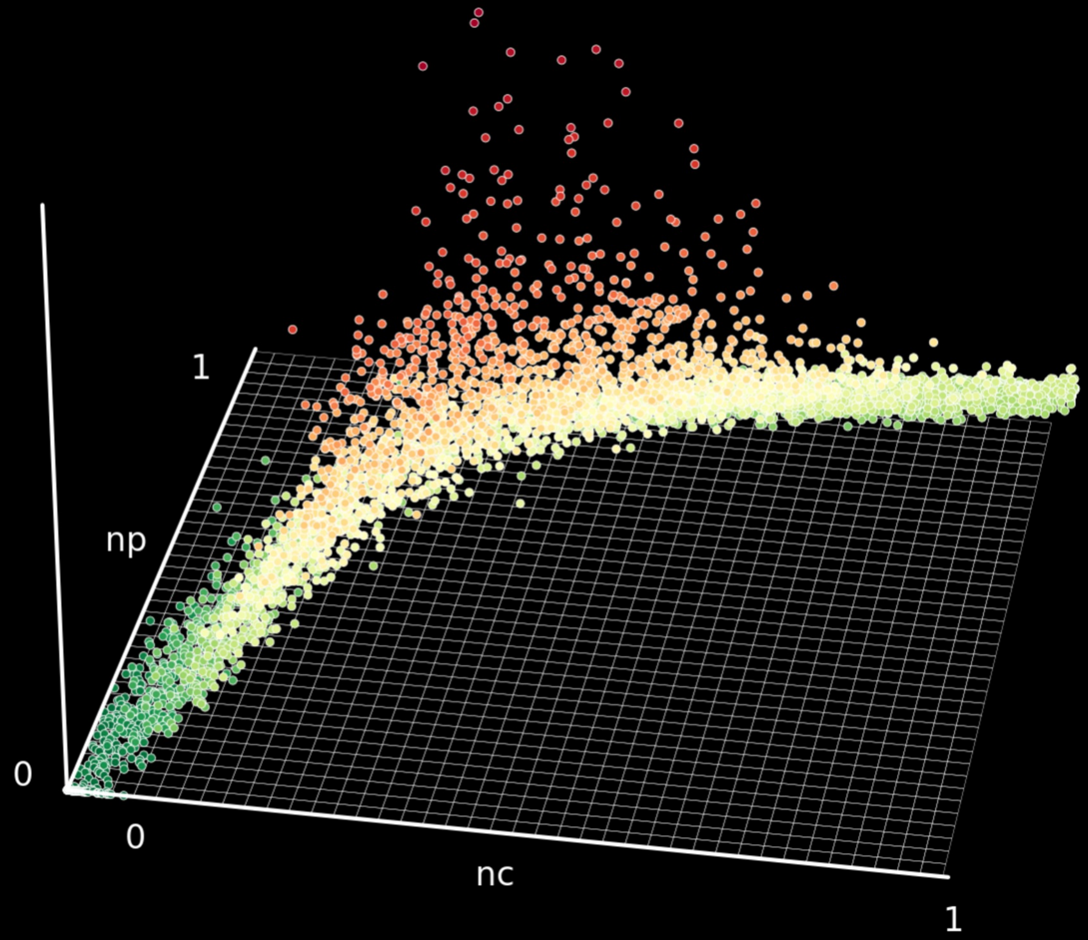
Where T^* = optimal value

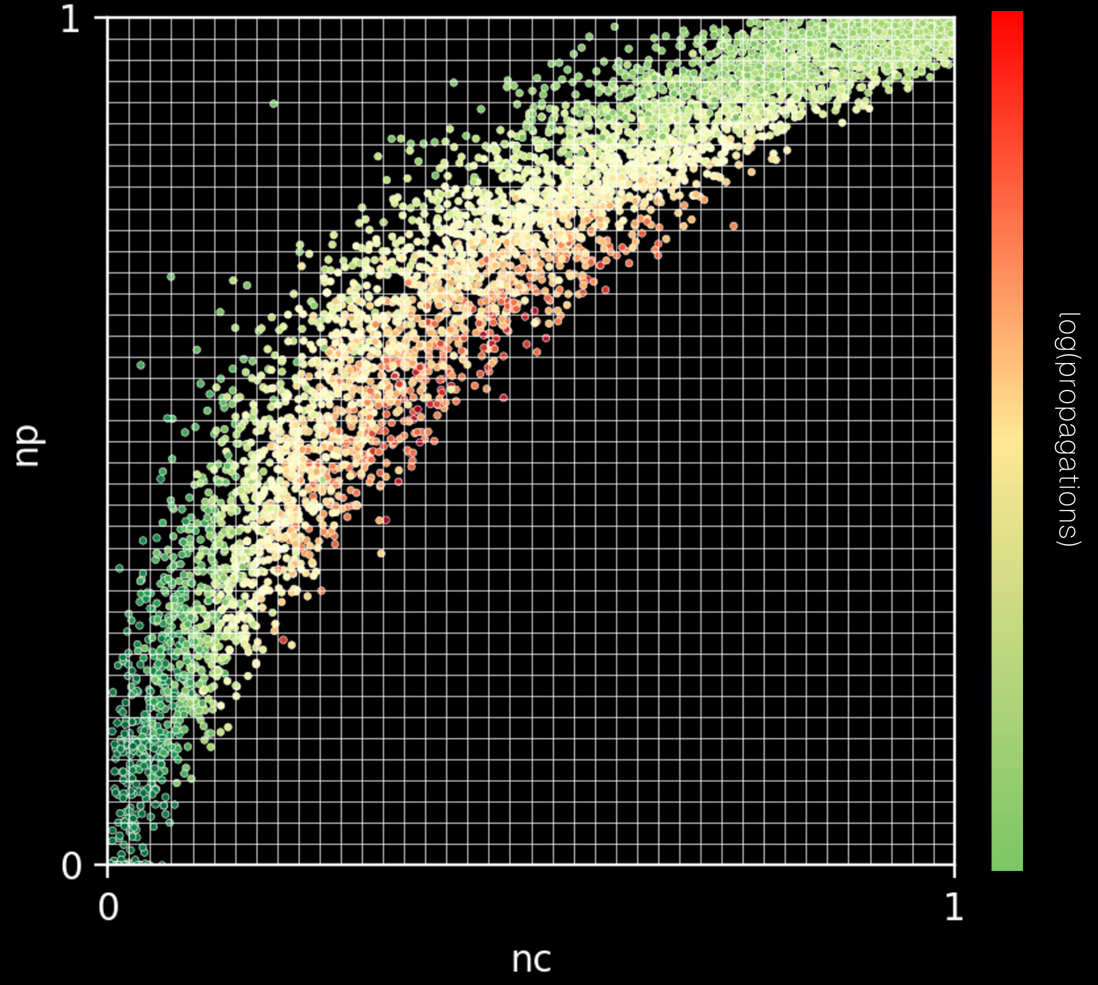


Where are the *really* hard problems?

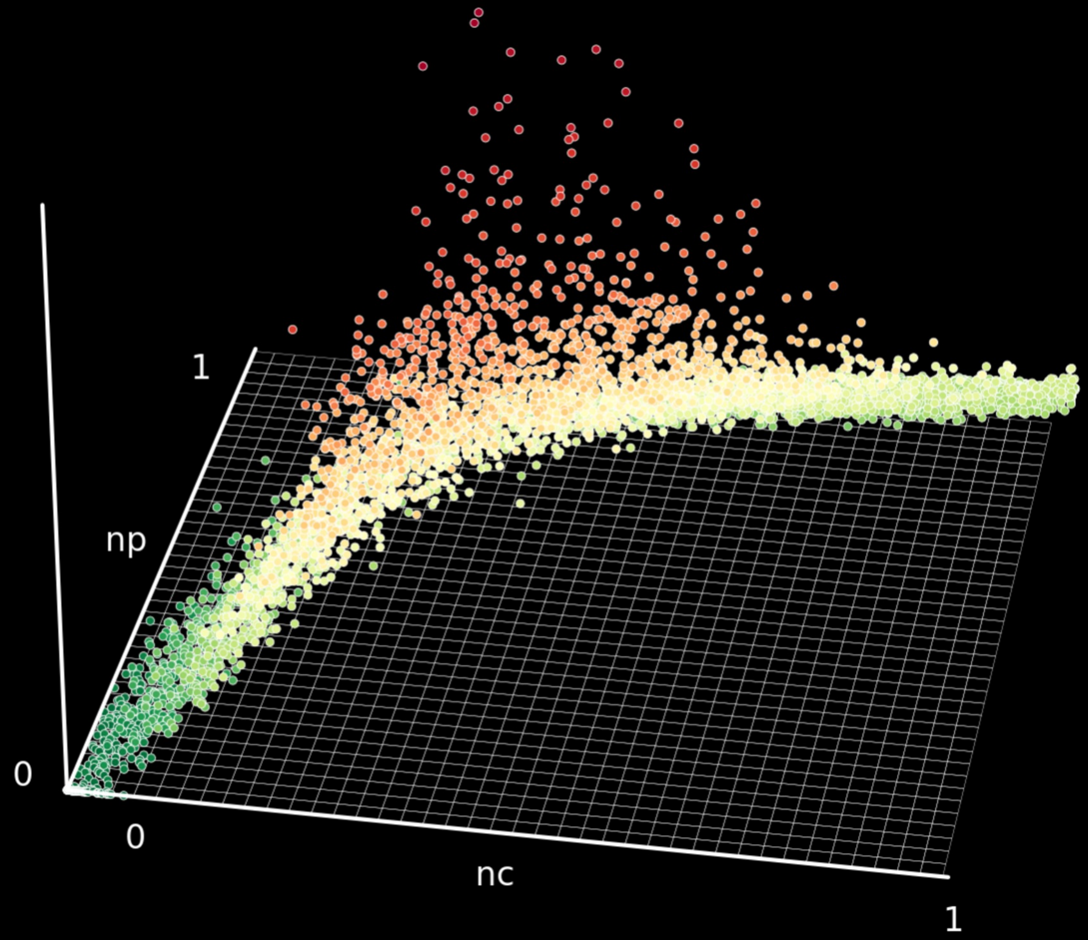


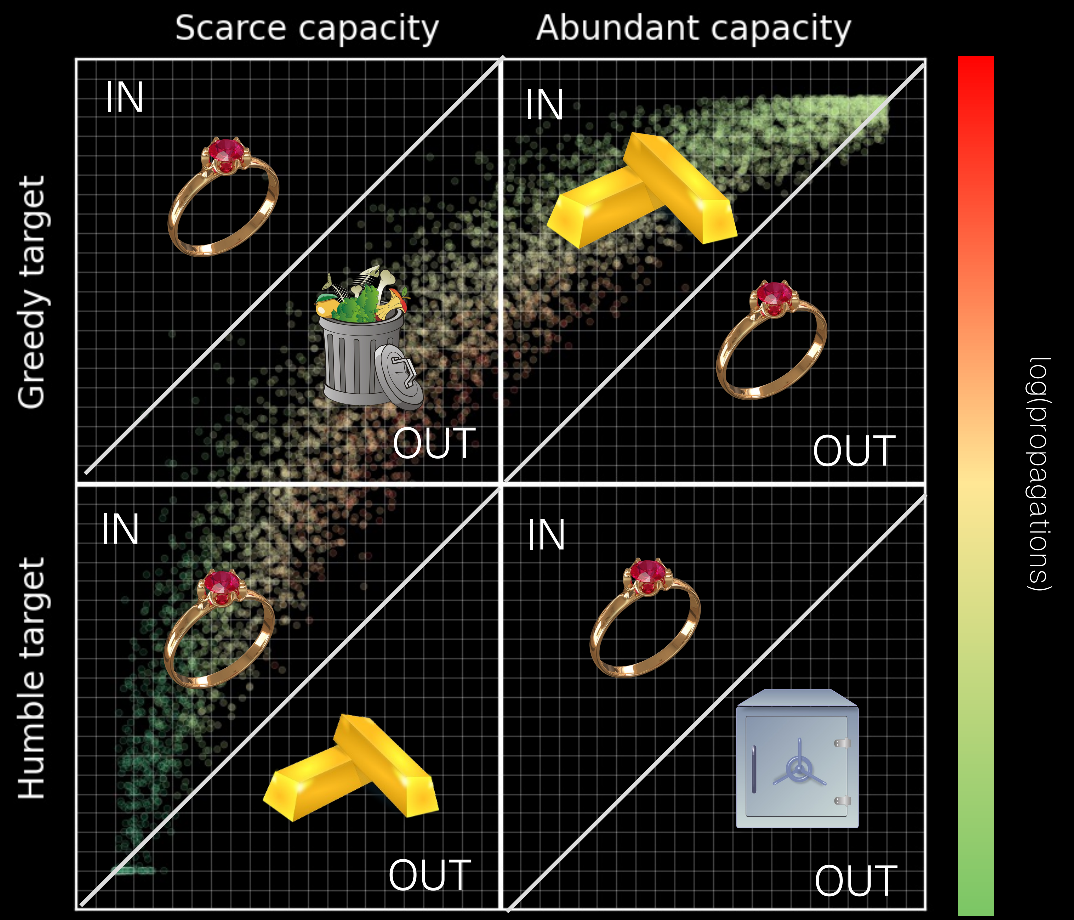
What happens in the empty space?



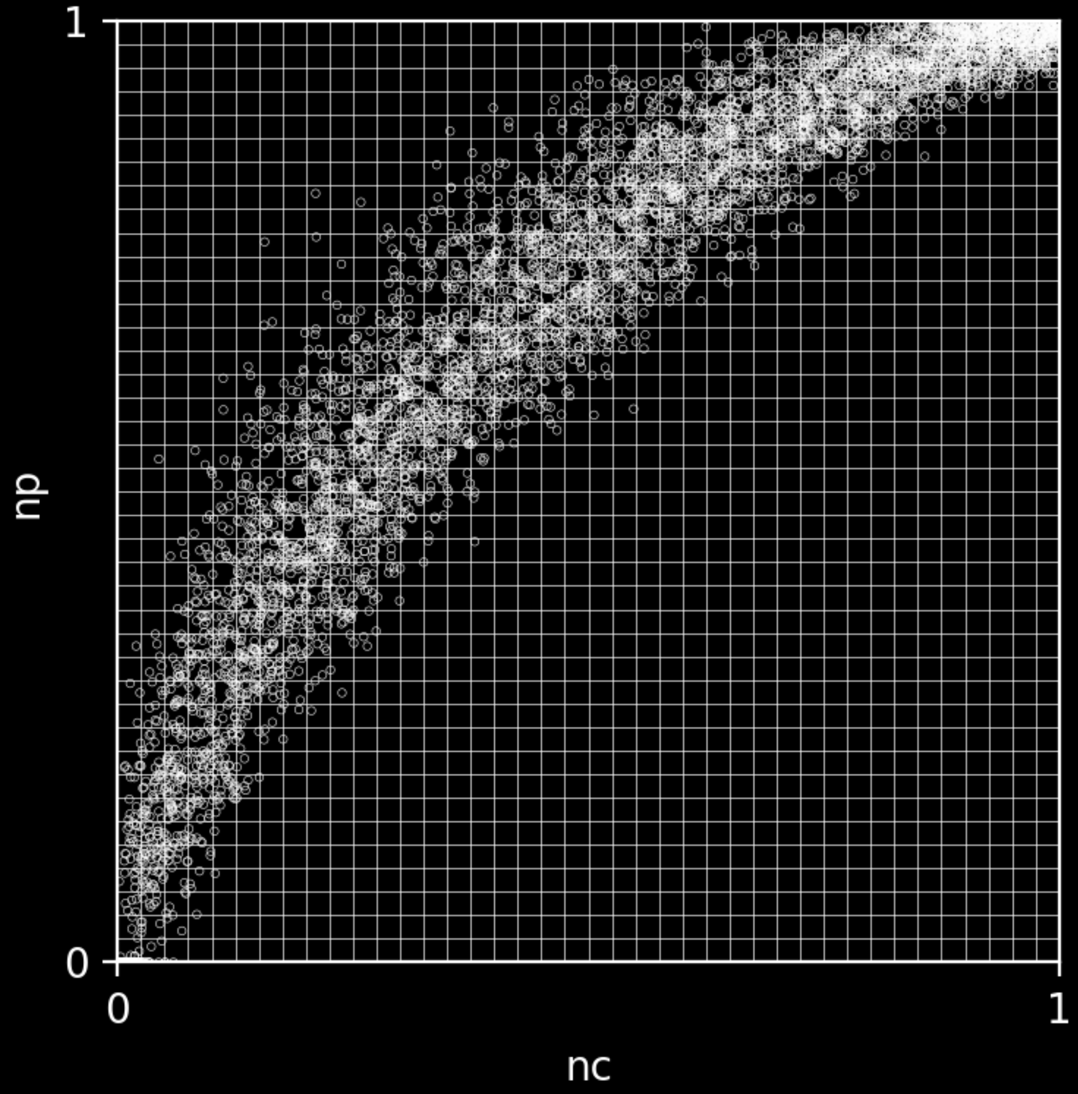


What happens in the empty space?

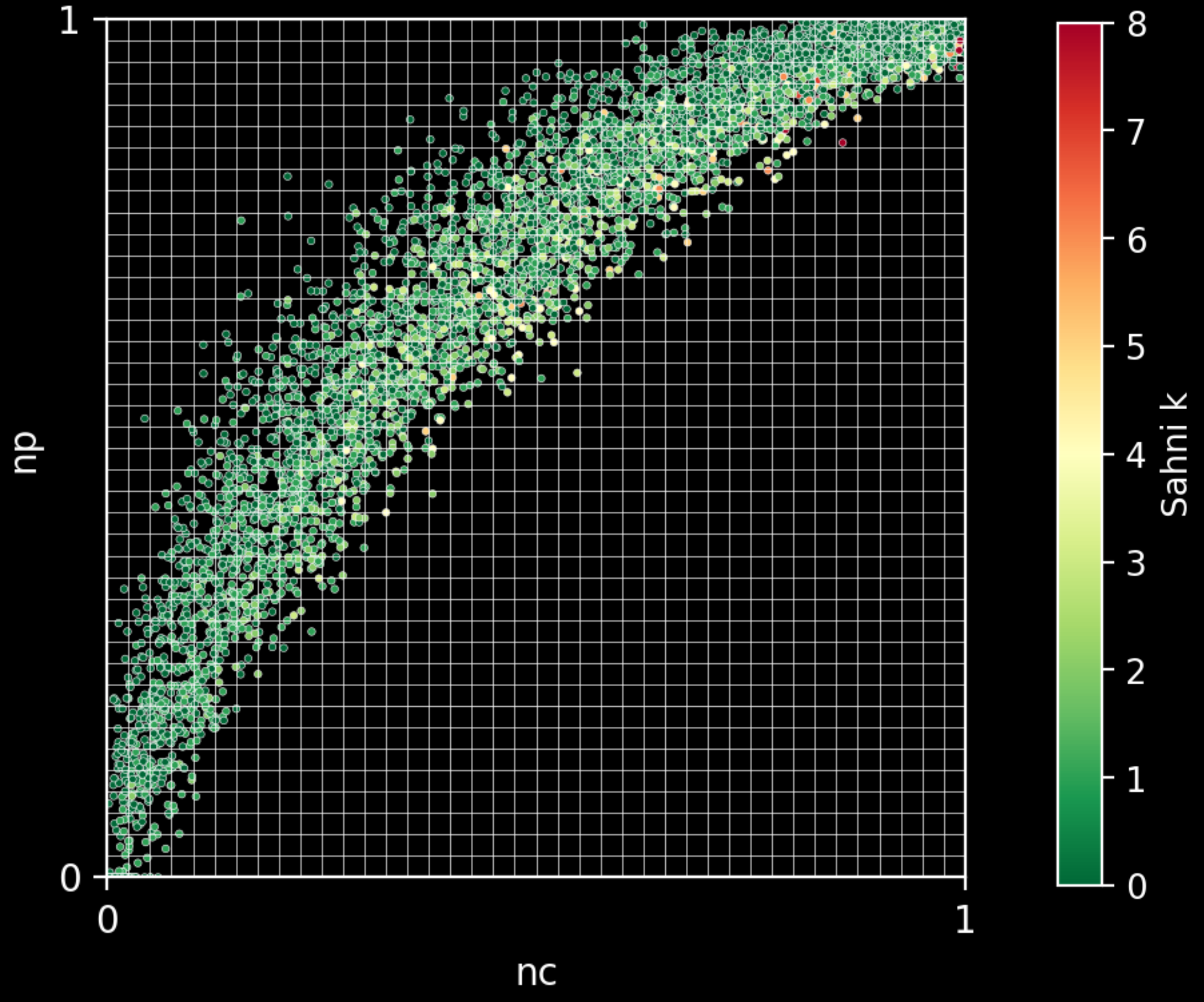




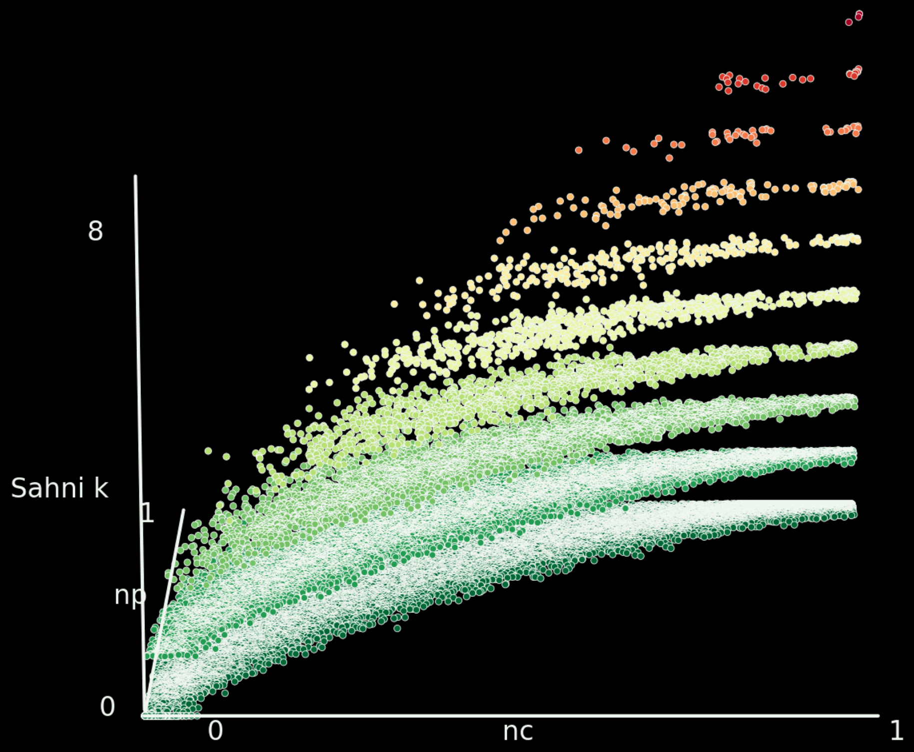
What about Sahni-k?



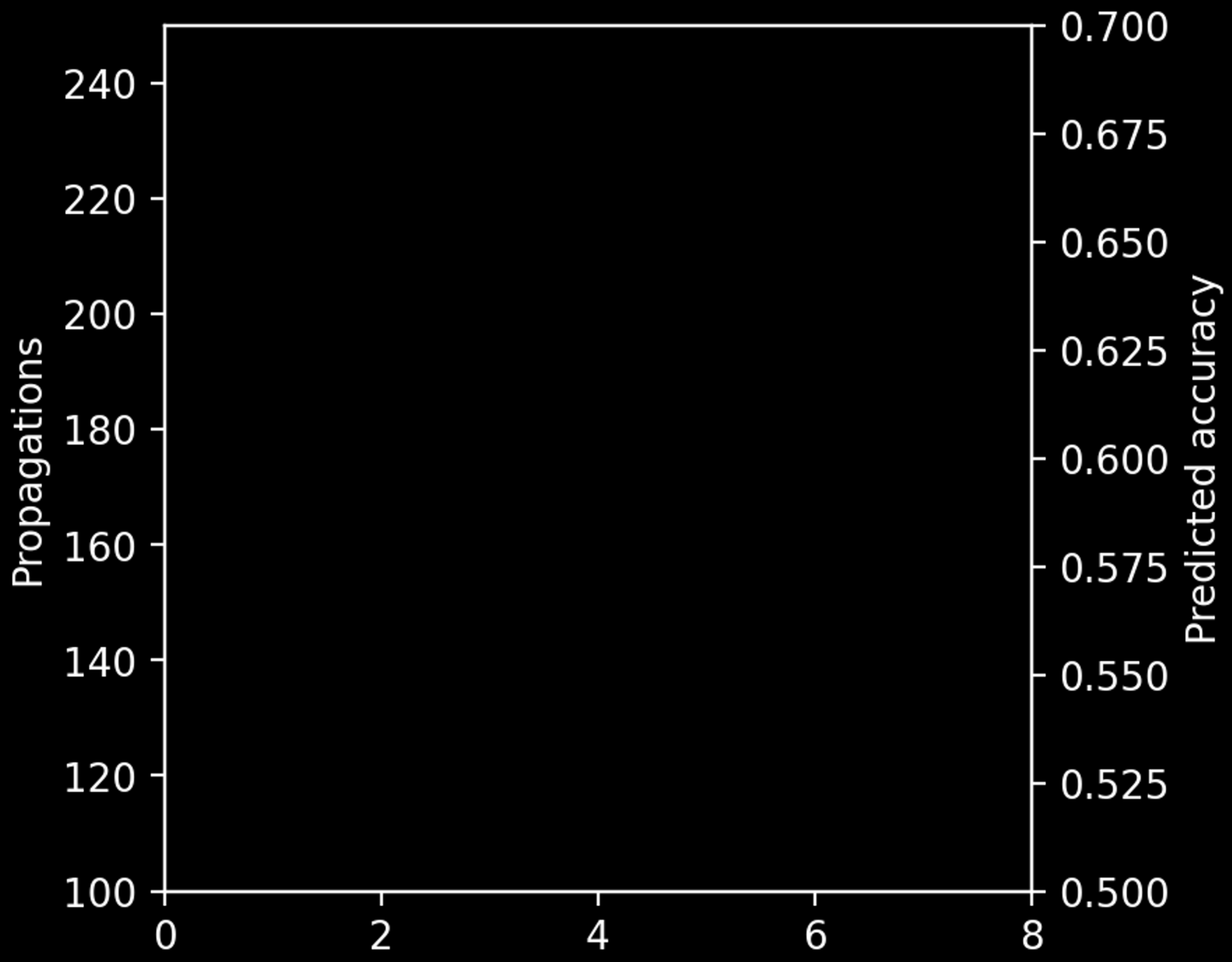
What about Sahni-k?



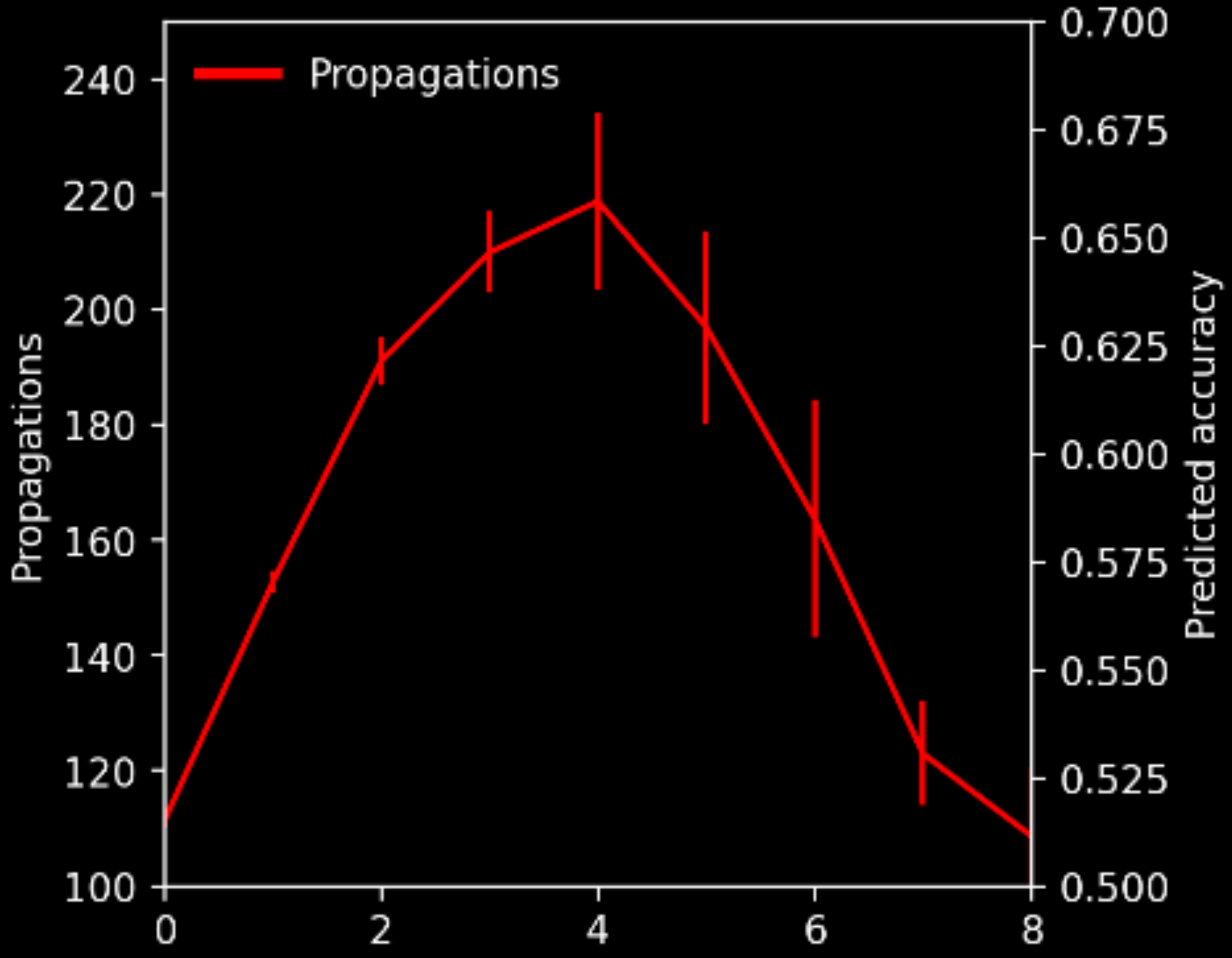
What about Sahni-k?



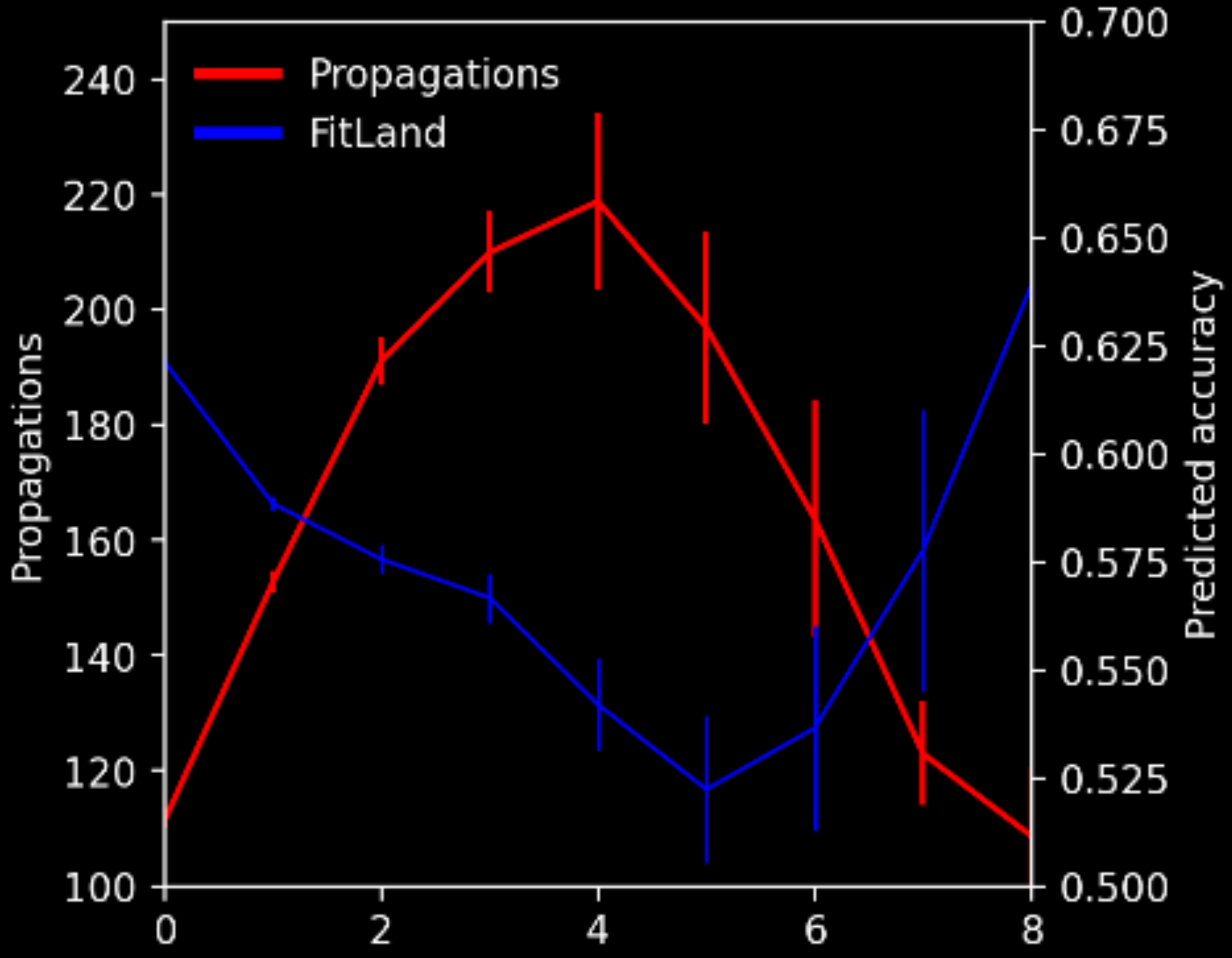
What about Sahni-k?



What about Sahni-k?



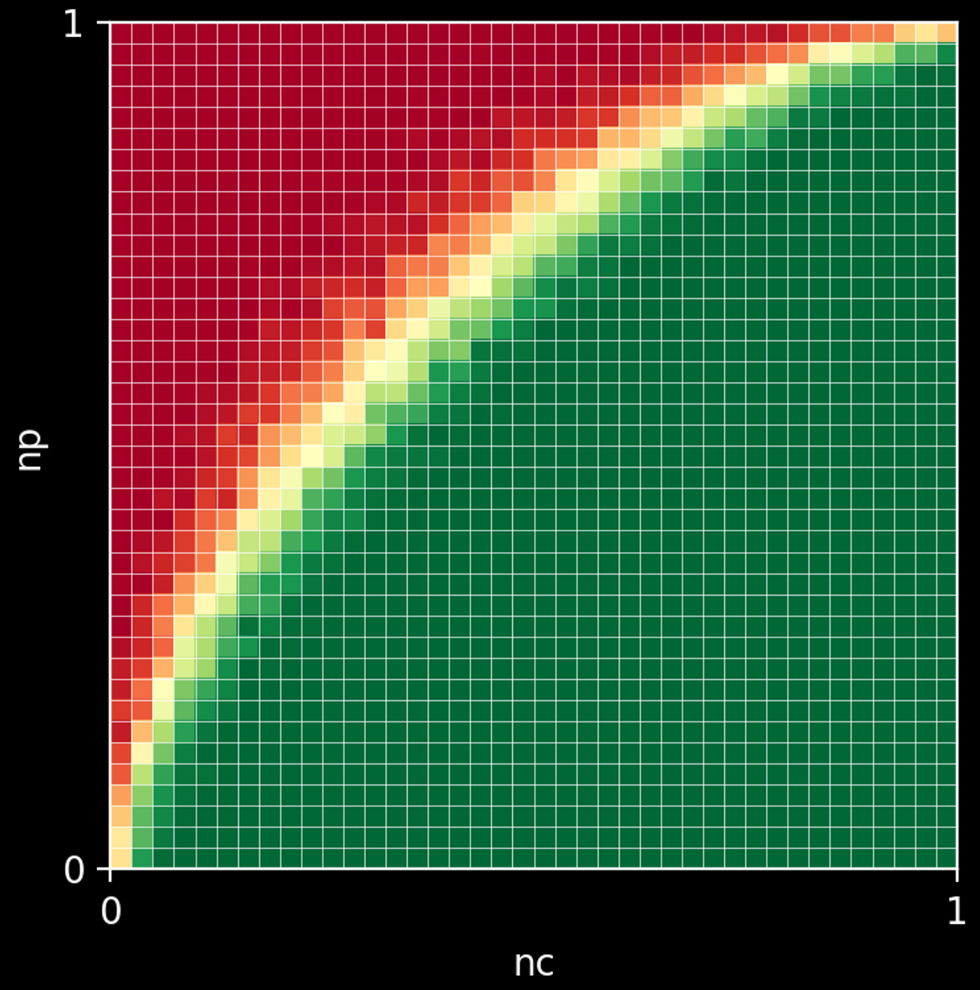
What about Sahni-k?



Summary

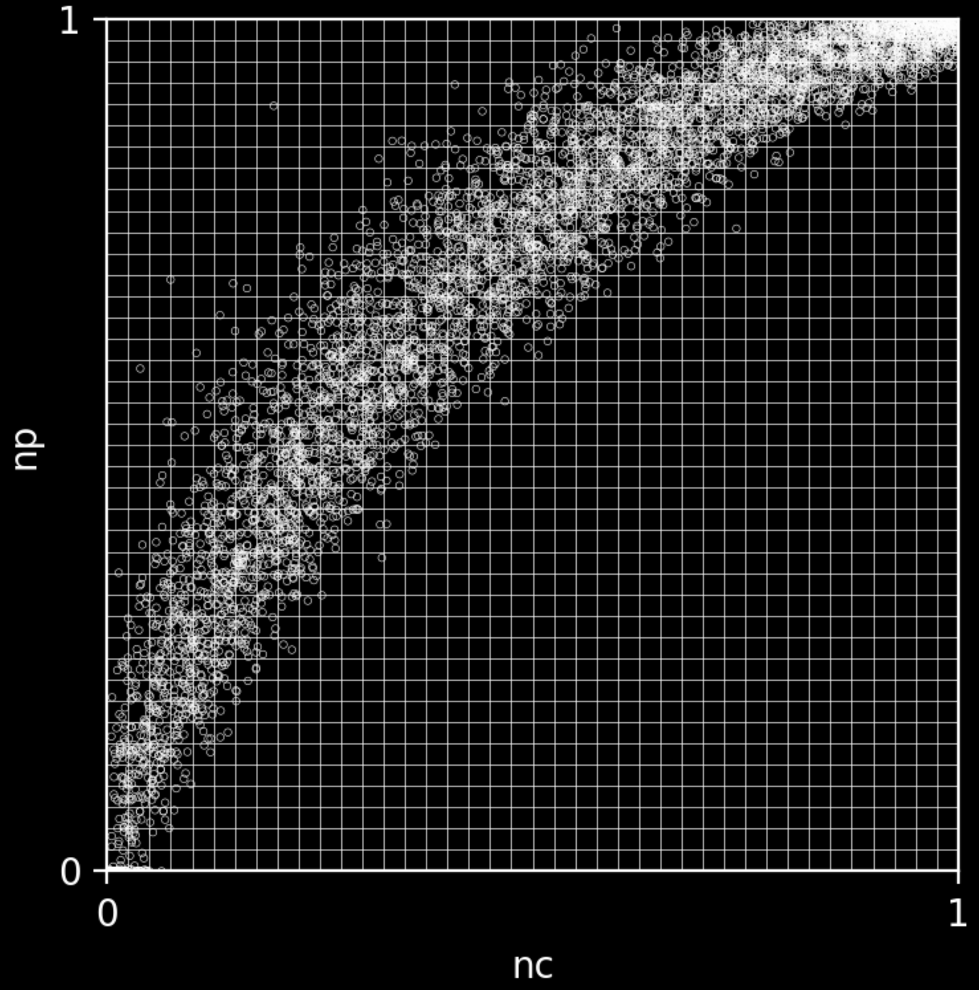
Summary

- Phase transitions exist in decision problems



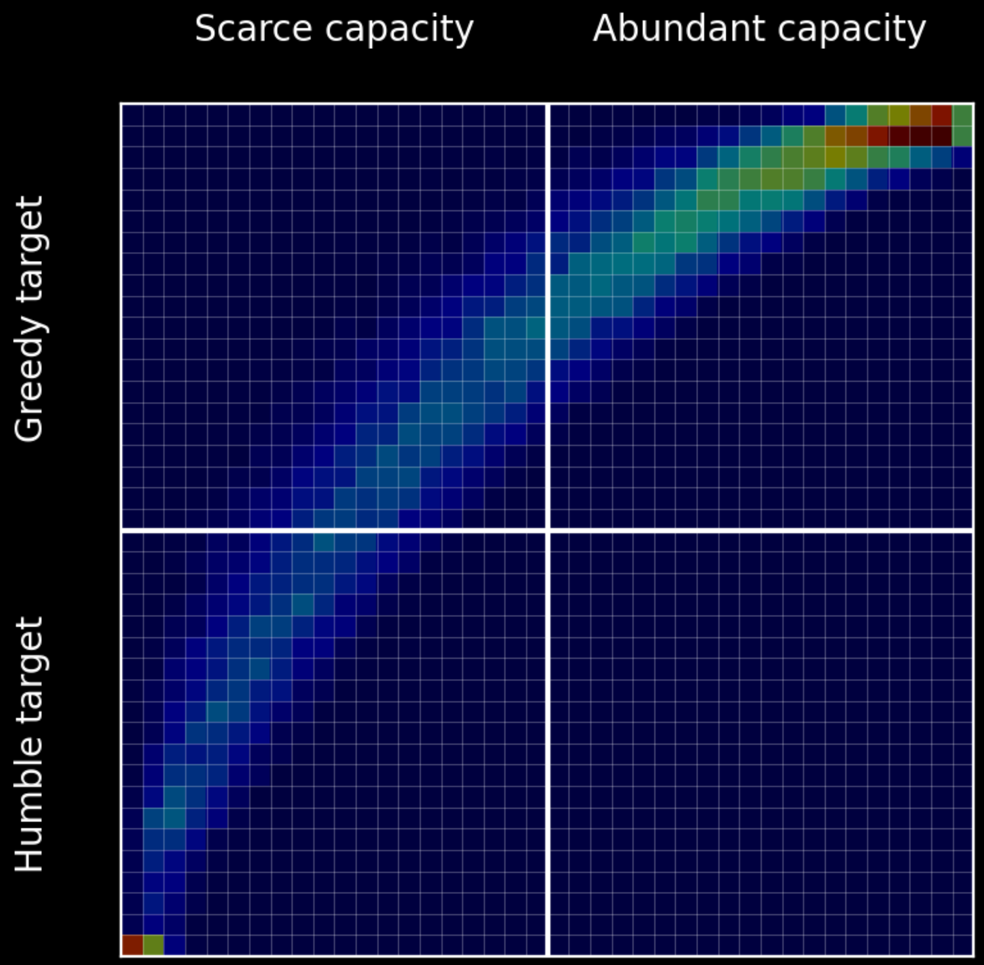
Summary

- Phase transitions exist in decision problems
- Order parameters can also be applied to optimisation instances



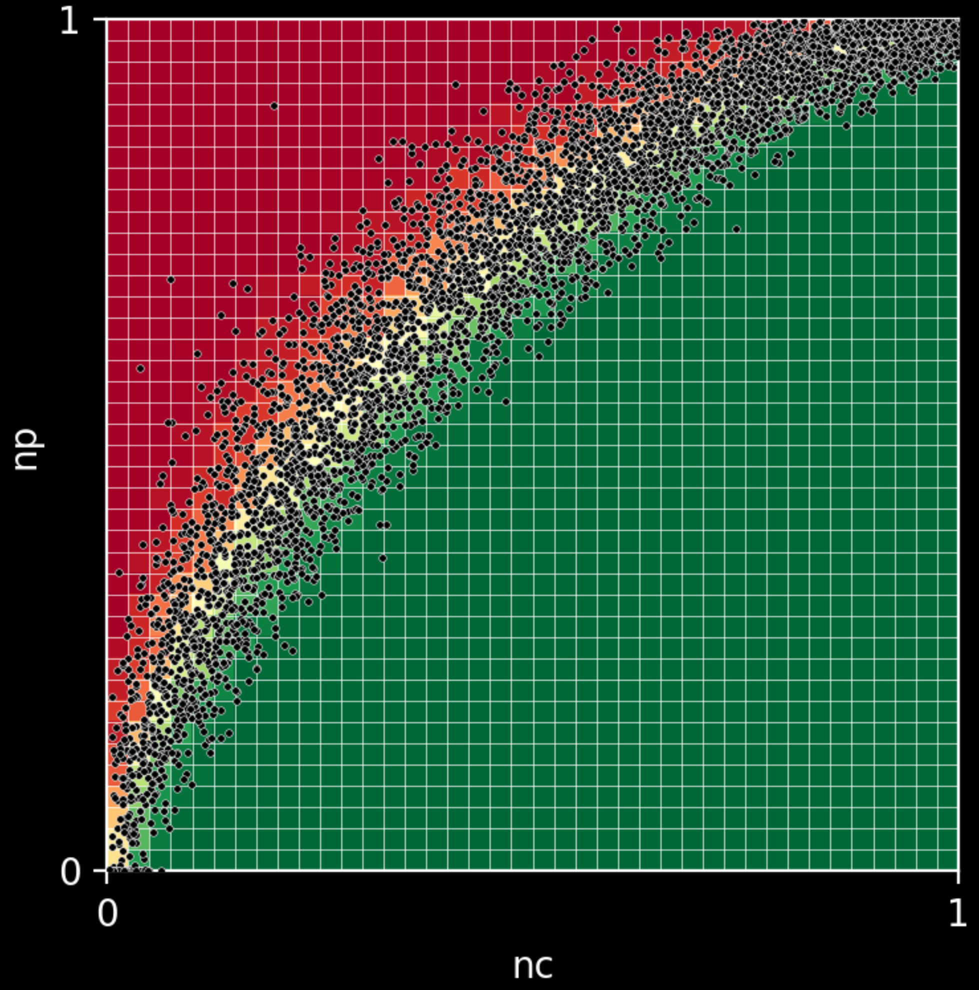
Summary

- Phase transitions exist in decision problems
- Order parameters can also be applied to optimisation instances
- Interesting (and intuitive) patterns emerge



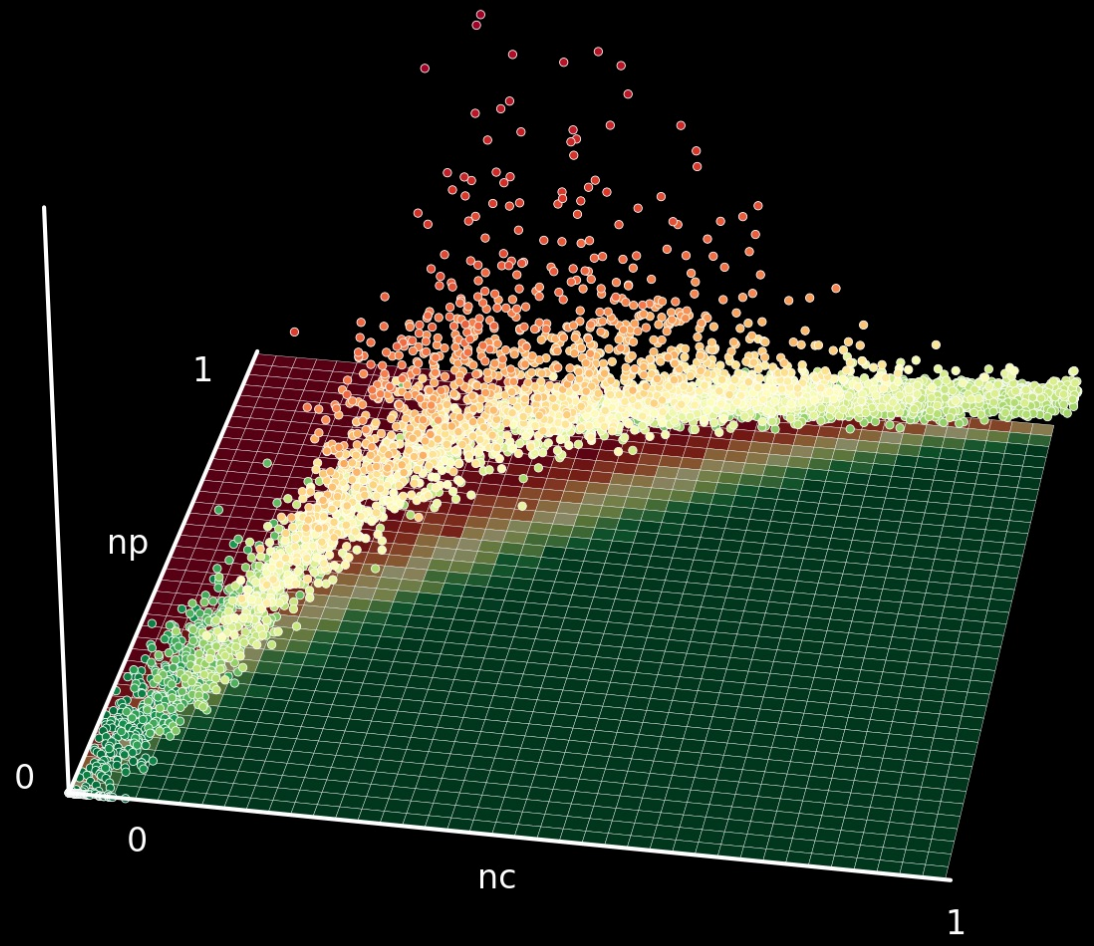
Summary

- Phase transitions exist in decision problems
- Order parameters can also be applied to optimisation instances
- Interesting (and intuitive) patterns emerge
  - Distribution of optimisation instances explains the decision phase transition



Summary

- Phase transitions exist in decision problems
- Order parameters can also be applied to optimisation instances
- Interesting (and intuitive) patterns emerge
  - Distribution of optimisation instances explains the decision phase transition
  - Order parameters predict computational complexity



Summary

- Phase transitions exist in decision problems
- Order parameters can also be applied to optimisation instances
- Interesting (and intuitive) patterns emerge
  - Distribution of optimisation instances explains the decision phase transition
  - Order parameters predict computational complexity
- Be very careful with Sahni-k.