GECCO

Genetic and Evolutionary Computation Conference

Madrid, Spain
July 11-15, 2015

On Evolutionary Approaches to Wind Turbine Placement with Geo-Constraints

July 13, 2015

Daniel Lückehe (Jade University of Applied Sciences), Markus Wagner (University of Adelaide), Oliver Kramer (University of Oldenburg)

GECCO 2015, Madrid

CARL VON OSSIETZKY universität OLDENBURG

JADEHOCHSCHULE Wilhelmshaven Oldenburg Elsfleth

SUB CRUCE LUMEN THE UNIVERSITY of ADELAIDE

Context

© www.adelaide.edu.au

- Objective:
  Maximize power output of renewable energy sources

- University of Adelaide
  - Adelaide: City in South Australia
  - A lot of solar and wind potential

© Carl von Ossietzky Universität Oldenburg

- University of Oldenburg and Jade University of Applied Sciences
  - Oldenburg: City in Lower Saxony, Northern Germany
  - Not so much solar potential, but a lot of wind

© iapg.jade-hs.de

- Focusing on wind turbines

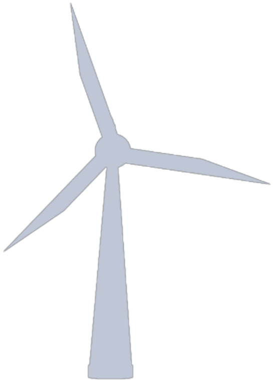
Motivation

- Behavior and effectiveness of wind turbines is strongly depending on their location
- ***Question: Where to place wind turbines to increase their power output?***
- Increase by:
  - Optimal positions caused by higher wind potential
  - Reduction of wake effects
- Modern Turbine, 40% full load hours, 0.1 €/kWh
  → 1 M€/a
- Even small improvements lead to large values

Content

1. Wind turbine placement scenario
   - How we calculate the power output of wind turbines?
2. Optimization
   - Fitness function definition
   - Solution representation
   - Optimization approaches
3. Experimental results

Wind



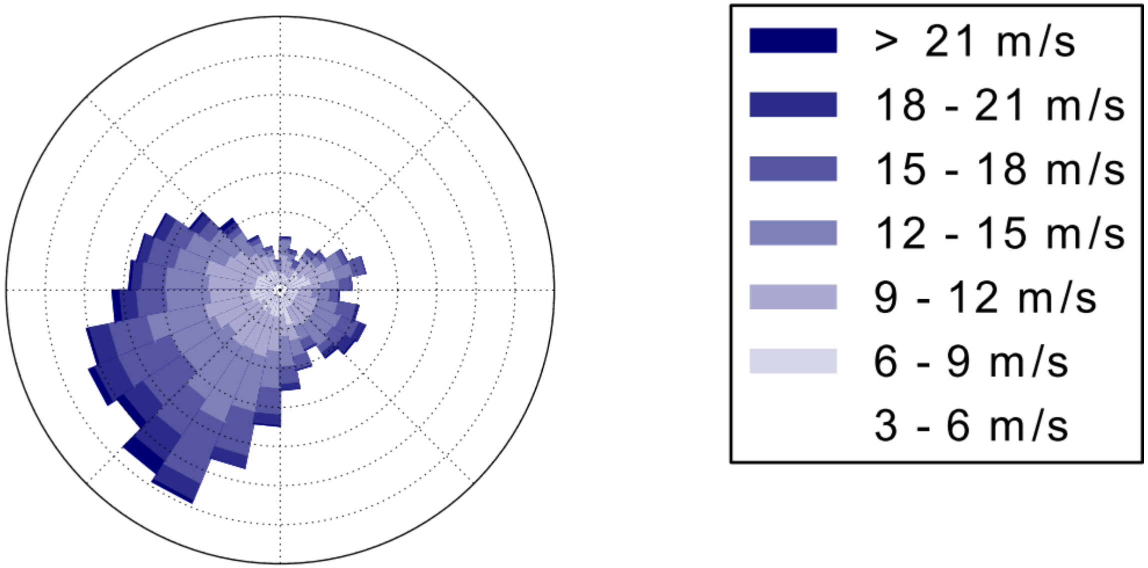
- Wind turbine
  - Produces power depending on wind speed at location
  - Power curve from Enercon E101
  - Constraint: Minimum distance between turbines

- Determination of the wind speed at location
  - Using COSMO-DE data from the German Weather Service (DWD)
  - Rotated grid over Germany (419 x 459 = 192,321, distance about 2.8 km)
  - Hourly wind vectors

  → more than 1.6 billion (419 x 459 x 365 x 24) wind vectors per year per height level

Wind Rose

- Rose: Distribution of wind speeds and directions
- Sort vectors by wind direction and location
- Wind rose for every grid position
- Bilinear interpolation using four grid positions

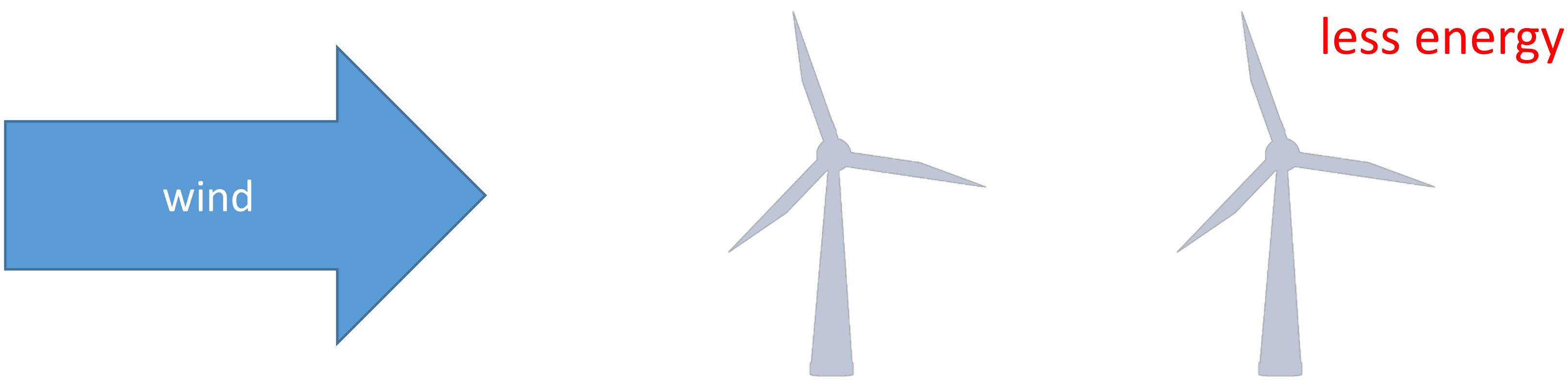


Wind Model

- Model based on wind distributions by Kusiak and Song (2010)
  - Weibull distribution to describe wind distributions
  - Discretization of wind speed and wind directions
  - Our modifications → see paper
- Model considers wake effects
  - Jensen wake model

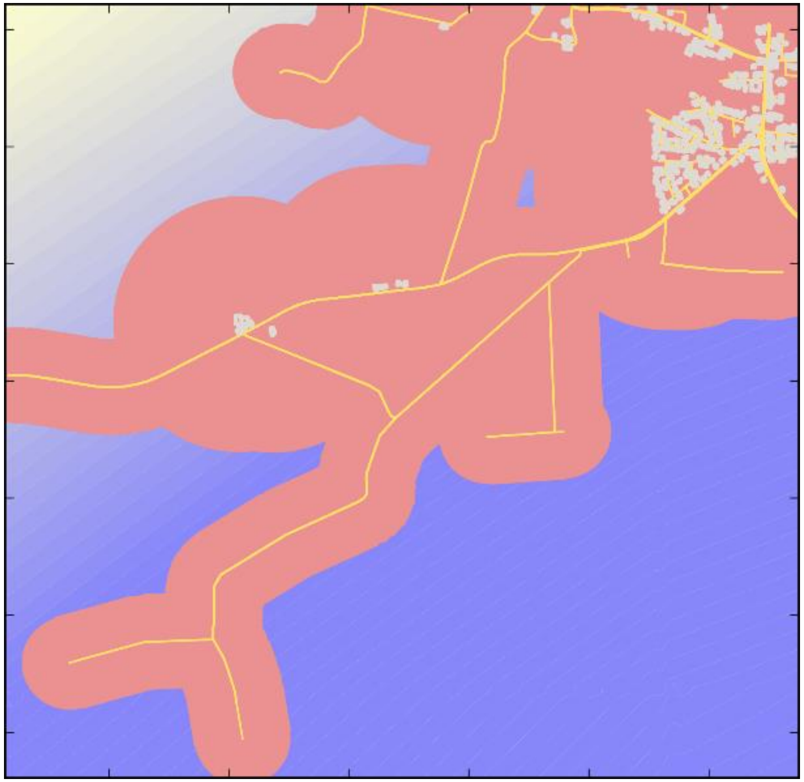
Wind Model

- Model based on wind distributions by Kusiak and Song (2010)
  - Weibull distribution to describe wind distributions
  - Discretization of wind speed and wind directions
  - Our modifications → see paper
- Model considers wake effects
  - Jensen wake model



Geographical Data

- Geographical data from OpenStreetMap
- Constraints for areas around
  - Buildings
  - Streets
- Constraint handling with death penalty
- Scenario 2 (from paper):
  - 3.3 x 3.3 km
  - 250 buildings
  - 64 streets consisting of 489 parts



Fitness Function

- Produced energy for a wind farm: sum of all turbines

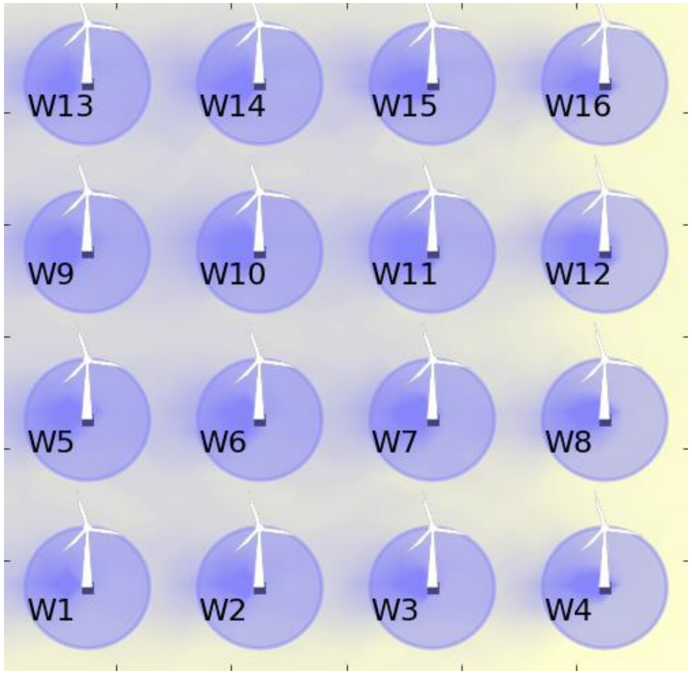
$$f(\mathbf{x}) = \sum_{i=1}^{N/2} E(\mathbf{t}_i)$$

- Fitness function depending on
  - Wind turbine E101 from Enercon
  - Wind data from COSMO-DE model
  - Power calculation based on Kusiak and Song with modifications
  - Jensen wake model
  - Geographical data from OpenStreetMap

Solution

- Solution $\boldsymbol{x}$ describes the positions of multiple turbines
  - e.g. $\boldsymbol{x} = [100, 300, 200, 200]$

- Initial solution:
  - Random
  - Chessboard pattern



- Circles:
  - Minimum distance between turbines

Evolutionary Strategies

- $\boldsymbol{x} = [100, 300, 200, 200]$
- Holistic approaches
  - Mutation will change every dimension of $\boldsymbol{x}$
- $\boldsymbol{x} = [100, 300, 200, 200]$
- Turbine-oriented approaches
  - Mutation will randomly pick the dimensions of one turbine

Evolutionary Strategies

- $\boldsymbol{x} = [100, 300, 200, 200]$

- Holistic approaches
  - Mutation will change every dimension of $\boldsymbol{x}$

- ***Adaptive*** $(\mathbf{1}+\mathbf{1})$***-ES***
  - ***Gaussian mutation to move turbines***
  - ***Rechenberg´s step size control***

Evolutionary Strategies

- $\boldsymbol{x} = [100, 300, 200, 200]$

- Holistic approaches
  - Mutation will change every dimension of $\boldsymbol{x}$

- Adaptive $(1 + 1)$-ES
- ***Covariance Matrix Adaptation Evolution Strategy (CMA-ES)***

Evolutionary Strategies

- $\boldsymbol{x} = [100, 300, 200, 200]$
- Turbine-oriented approaches
  - Mutation will randomly pick the dimensions of one turbine
- ***Adaptive*** $\mathbf{(1+1)}$***-ES***
  - ***Gaussian mutation***
  - ***Special Case: 1 dimension***
  - ***Rechenberg´s step size control***

Evolutionary Strategies

- $\boldsymbol{x} = [100, 300, 200, 200]$
- Turbine-oriented approaches
  - Mutation will randomly pick the dimensions of one turbine
- Adaptive $(1+1)$-ES (1 dim.)
- ***Replacing***

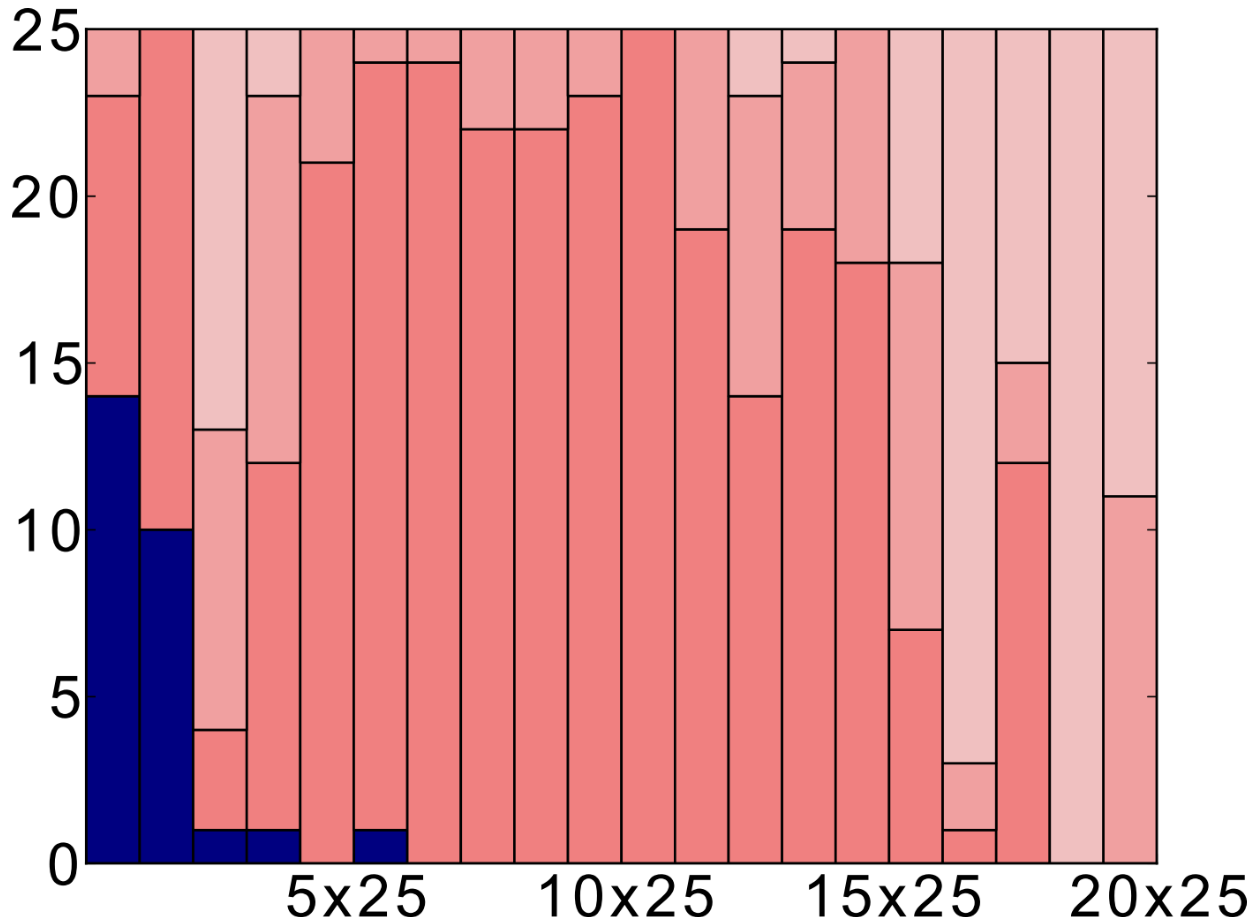
Evolutionary Strategies

- $\boldsymbol{x} = [100, 300, 200, 200]$
- Turbine-oriented approaches
  - Mutation will randomly pick the dimensions of one turbine
- Adaptive $(1+1)$-ES (1 dim.)
- Replacing
- ***Deterministic*** $\mathbf{(1+1)}$***-ES***
  - ***Gaussian mutation***
  - ***Starting with step size equal the map size***
  - ***Decreased*** **linearly**
  - ***Ending with step size: 1 / "fitness function calls" x map size***

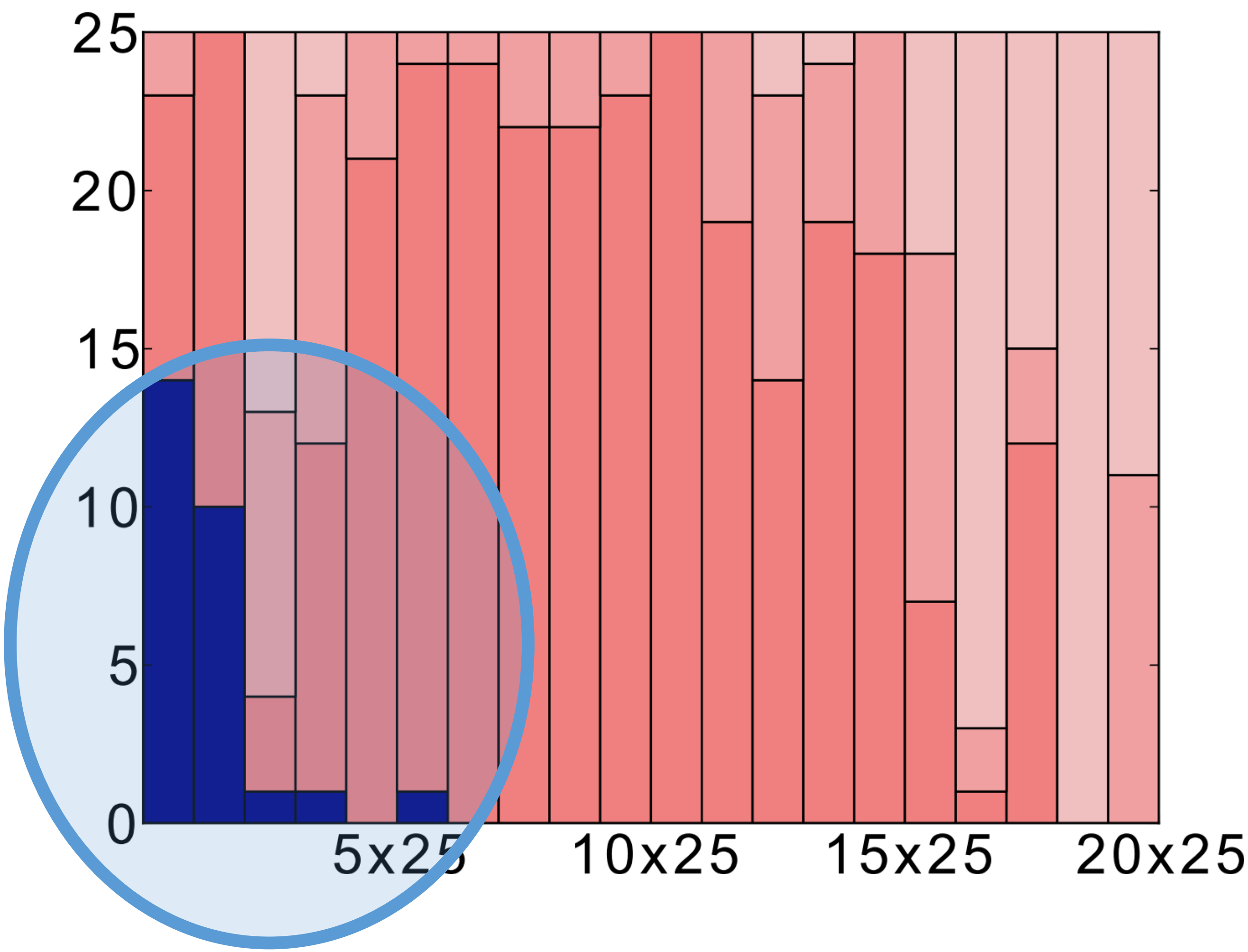
Evolutionary Strategies

- $\boldsymbol{x} = [100, 300, 200, 200]$
- Turbine-oriented approaches
  - Mutation will randomly pick the dimensions of one turbine
- Adaptive $(1+1)$-ES (1 dim.)
- Replacing
- Deterministic $(1+1)$-ES
- ***Self-Adaptive*** $(\mathbf{1}+\boldsymbol{\lambda})$***-ES***
  - ***Choose between replace and Gaussian mutation***
  - ***Can also mutate multiple turbines in one generation***
  - ***Operation and number of turbines are controlled self-adaptively***

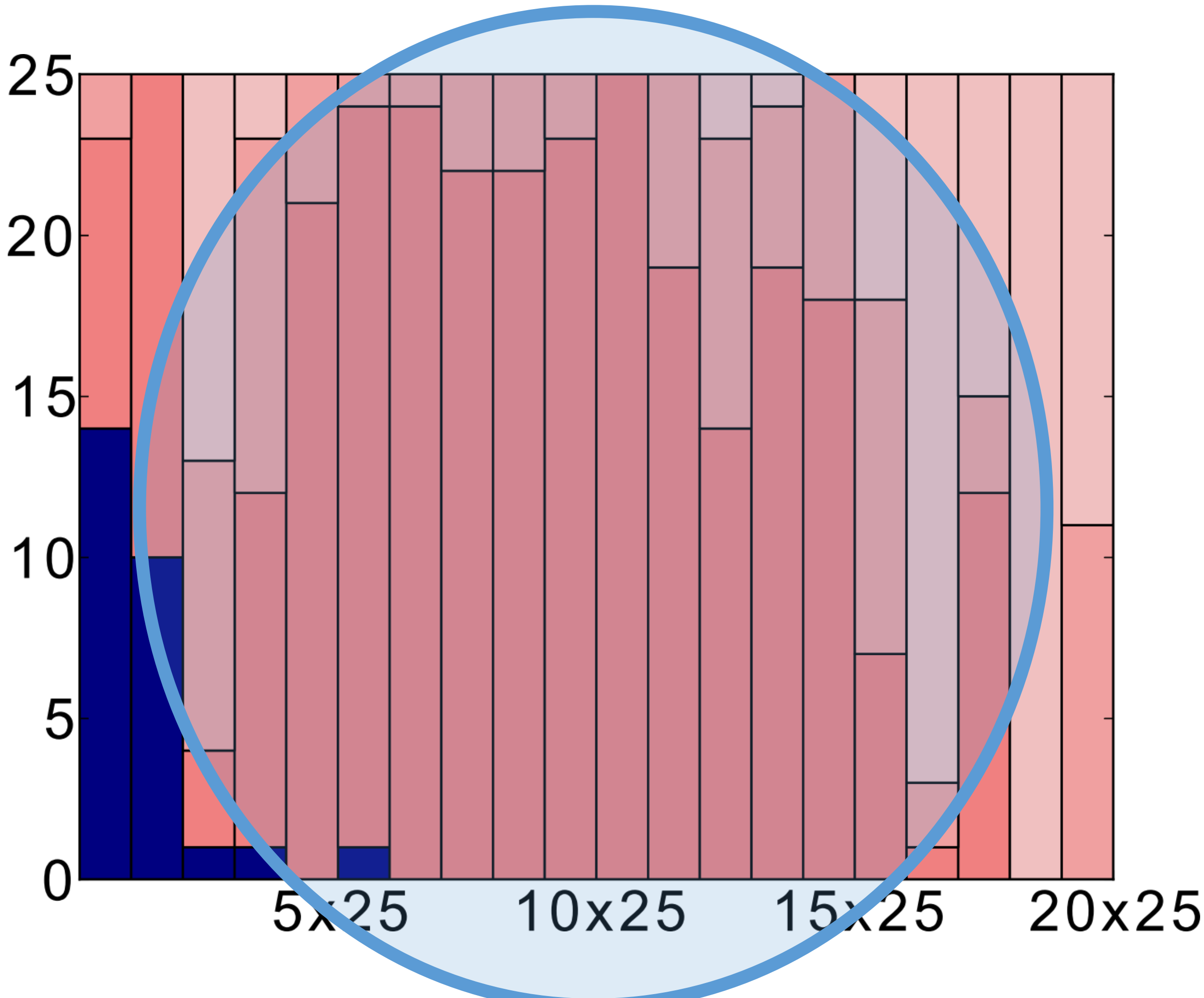
Analysis of Operator Ratios



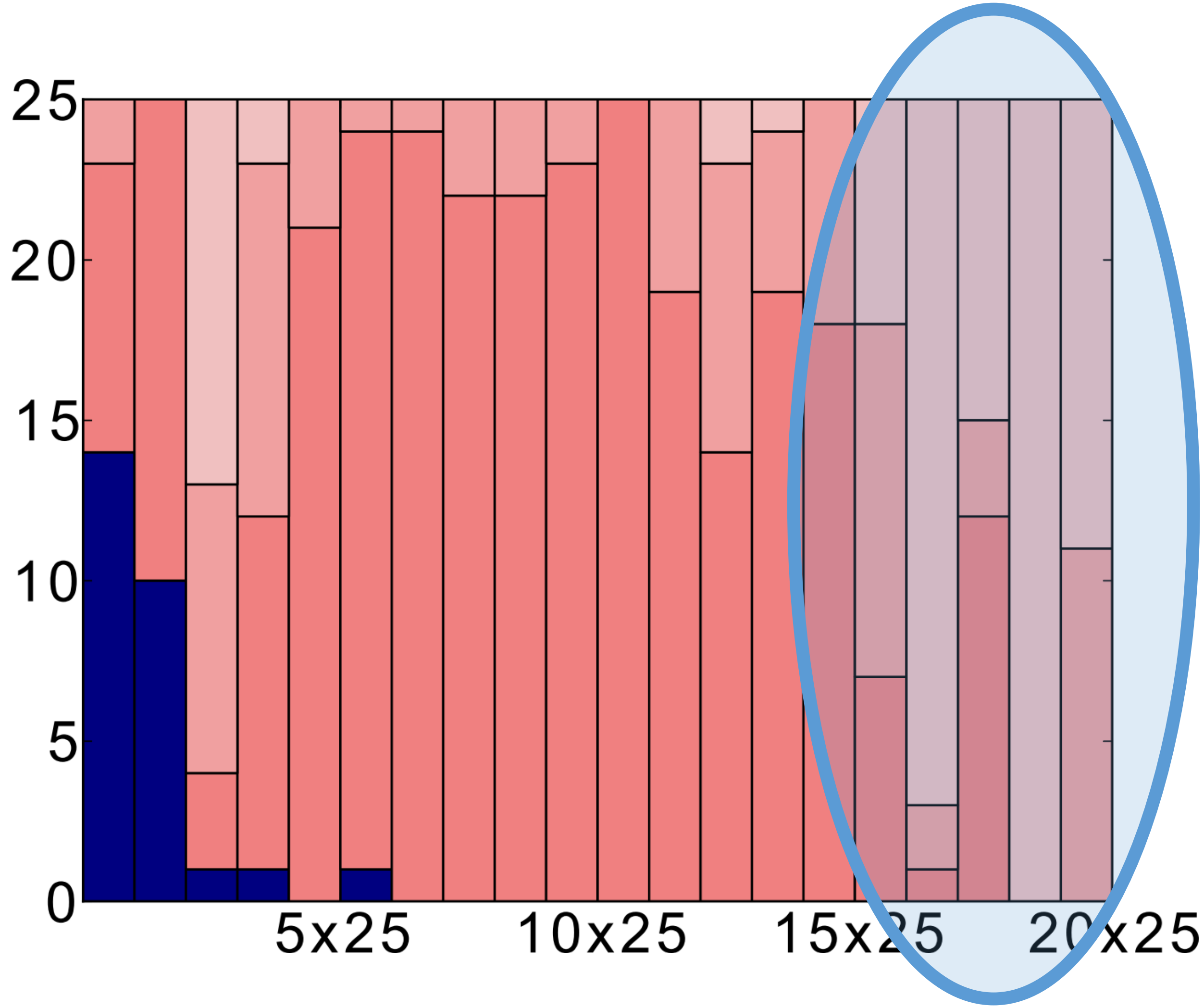
Analysis of Operator Ratios



Analysis of Operator Ratios



Analysis of Operator Ratios



Comparison of Evolutionary Algorithms

- Scenario 2: 36-dimensional optimization problem (18 turbines) (results for random initialization in paper)

Init.	Chess	
Algo.	Mean ± Std	Max
Init.	10944.2 ± 0.0	10944.2
$(1+1)^N$	11221.3 ± 33.0	11287.3
CMA	11359.7 ± 12.5	11386.6
$(1+1)^1$	11399.3 ± 17.5	11444.5
Rep.	11484.0 ± 9.7	11505.0
$(1+1)^{\mathbf{t}}$	**11524.1** ± 8.4	**11538.2**
$(1+\lambda)$	11516.5 ± 11.7	11537.2

Comparison of Evolutionary Algorithms

- Scenario 2: 36-dimensional optimization problem (18 turbines) (results for random initialization in paper)

Init.	Chess	
Algo.	Mean $\pm$ Std	Max
Init.	10944.2 ± 0.0	10944.2
$(1{+}1)^N$	11221.3 ± 33.0	11287.3
CMA	11359.7 ± 12.5	11386.6
$(1{+}1)^1$	11399.3 ± 17.5	11444.5
Rep.	11484.0 ± 9.7	11505.0
$(1{+}1)^{\mathbf{t}}$	$\mathbf{11524.1} \pm 8.4$	**11538.2**
$(1{+}\lambda)$	11516.5 ± 11.7	11537.2

Comparison of Evolutionary Algorithms

- Scenario 2: 36-dimensional optimization problem (18 turbines) (results for random initialization in paper)

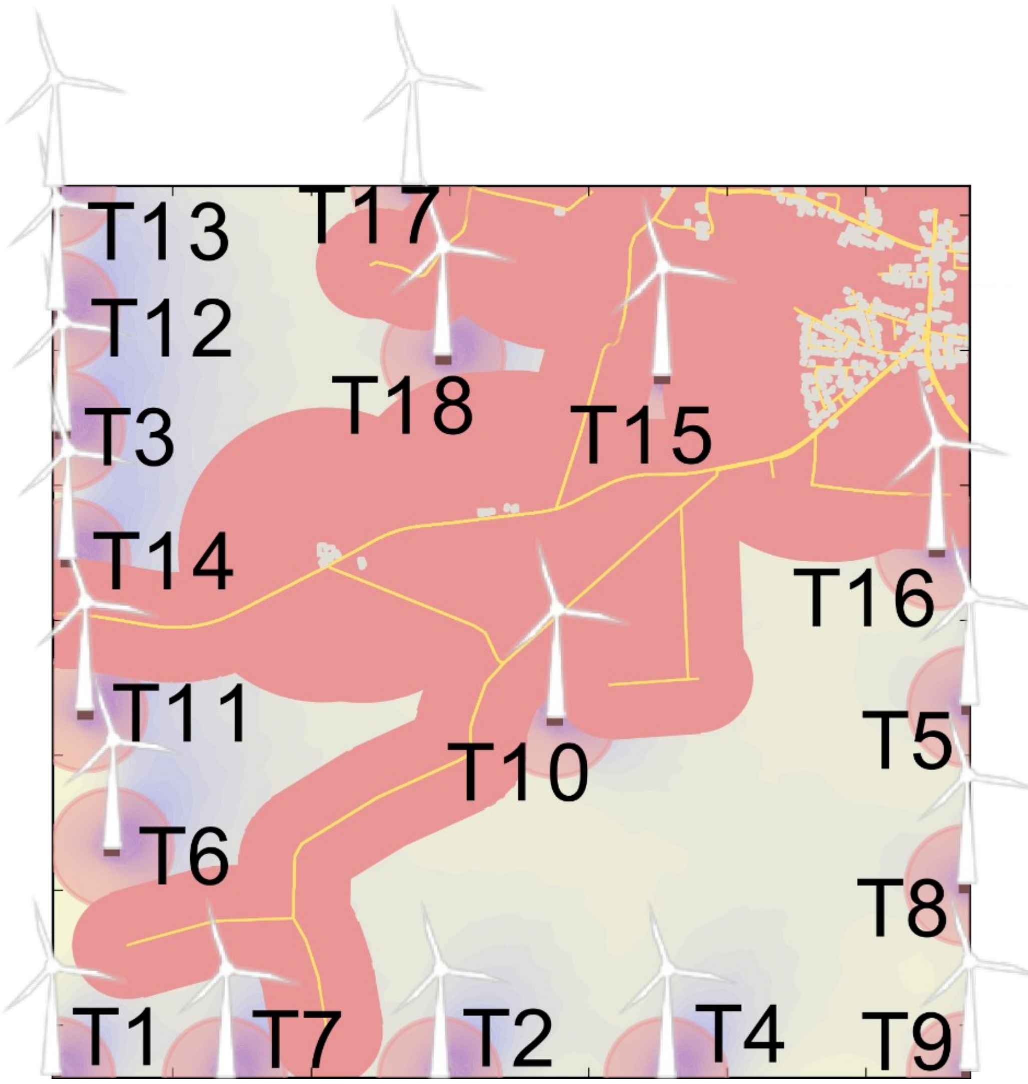
Init.	Chess	
Algo.	Mean ± Std	Max
Init.	10944.2 ± 0.0	10944.2
$(1+1)^N$	11221.3 ± 33.0	11287.3
CMA	11359.7 ± 12.5	11386.6
$(1+1)^1$	11399.3 ± 17.5	11444.5
Rep.	11484.0 ± 9.7	11505.0
$(1+1)^{\mathbf{t}}$	**11524.1** ± 8.4	**11538.2**
$(1+\lambda)$	11516.5 ± 11.7	11537.2

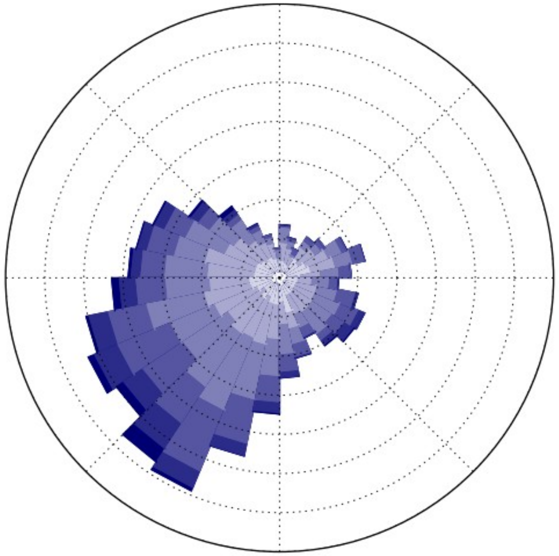
Comparison of Evolutionary Algorithms

- Scenario 2: 36-dimensional optimization problem (18 turbines) (results for random initialization in paper)

Init.	Chess	
Algo.	Mean $\pm$ Std	Max
Init.	10944.2 ± 0.0	10944.2
$(1+1)^N$	11221.3 ± 33.0	11287.3
CMA	11359.7 ± 12.5	11386.6
$(1+1)^1$	11399.3 ± 17.5	11444.5
Rep.	11484.0 ± 9.7	11505.0
$(1+1)^{\mathbf{t}}$	$\mathbf{11524.1} \pm 8.4$	**11538.2**
$(1+\lambda)$	11516.5 ± 11.7	11537.2

Best Turbine Placement Result





Conclusion & Future Work

Conclusion

- Wind turbine placement leads to different optimization problem
- Self-adaptive approach first replaces, **then** moves turbines
- Turbine-oriented approaches outperform holistic approaches

Conclusion & Future Work

Conclusion

- Wind turbine placement leads to different optimization problem
- Self-adaptive approach first replaces, **then** moves turbines
- Turbine-oriented approaches outperform holistic approaches

Future Work

- Advanced contraint handling methods
- Add more features to the model

Conclusion & Future Work

Conclusion

- Wind turbine placement leads to different optimization problem
- Self-adaptive approach first replaces, **then** moves turbines
- Turbine-oriented approaches outperform holistic approaches

Future Work

- Advanced contraint handling methods
- Add more features to the model

Thank you!

CARL VON OSSIETZKY universität OLDENBURG

JADEHOCHSCHULE Wilhelmshaven Oldenburg Elsfleth

SUB CRUCE LUMEN THE UNIVERSITY of ADELAIDE

GECCO

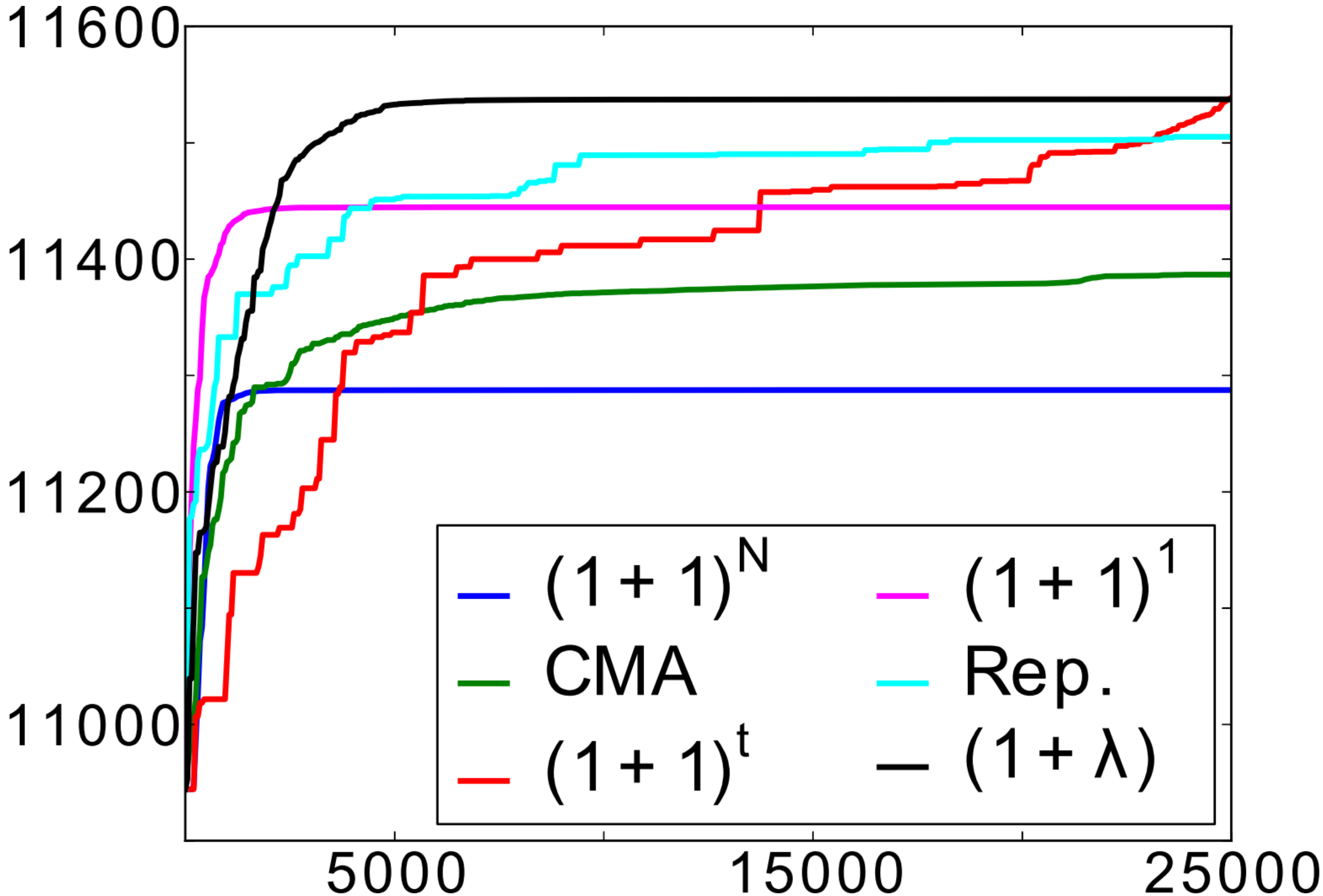
Genetic and Evolutionary Computation Conference

Madrid, Spain
July 11-15, 2015

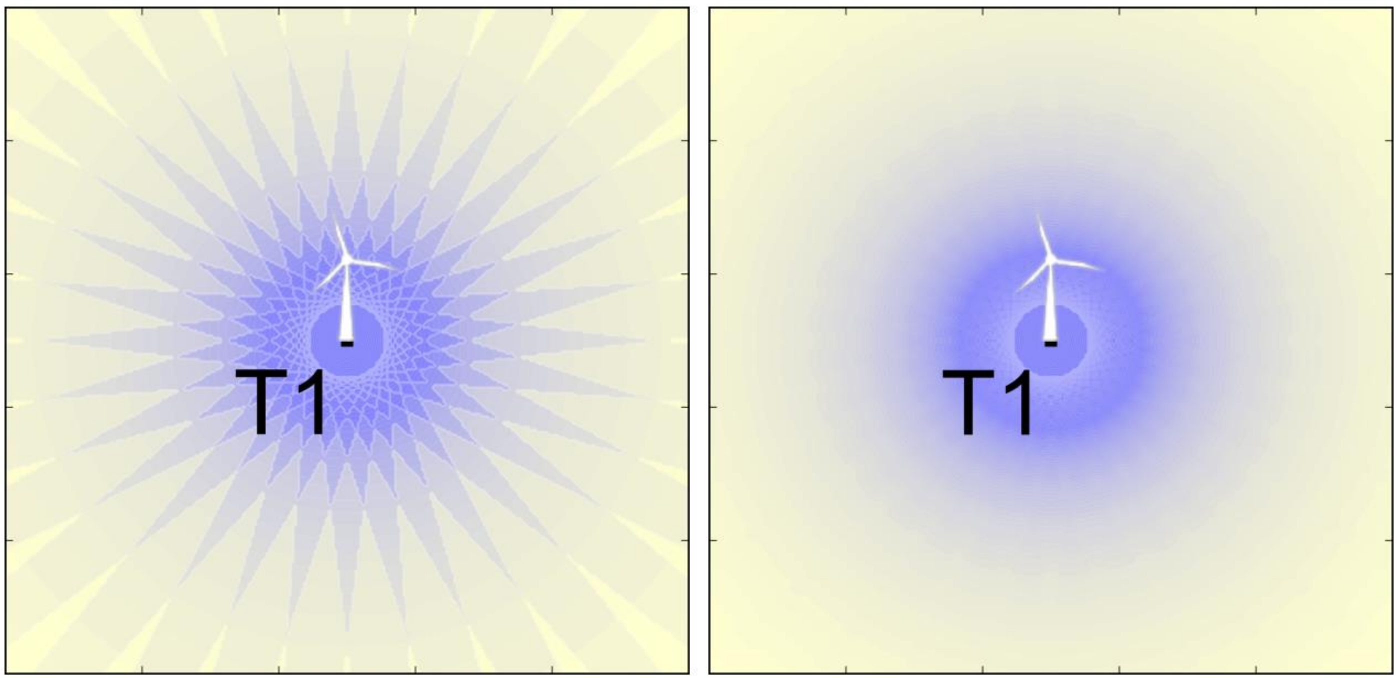
Context

- This work follows the project *EnerGeoPlan*
  - 2011 – 2013
  - Objective: *To bring energy supply, spatial planning, and grid planning together*
  - Funded by the Ministry for Science and Culture of the State of Lower Saxony
  - Co-operation partners: *Gemeinde Ganderkesee, EWE Netz GmbH*
  - *http://www.offis.de/f_e_bereiche/verkehr/projekt/projekte/energeoplan.html*

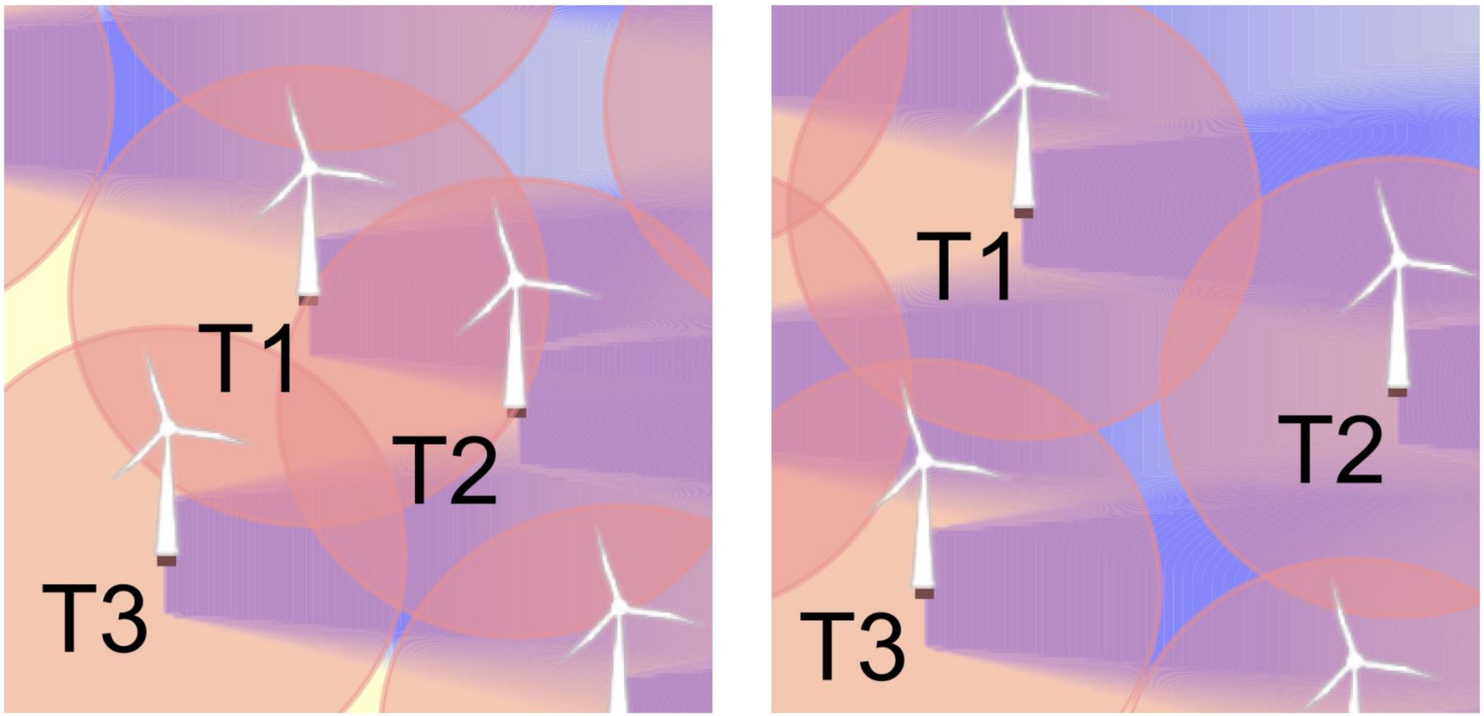
Evolutionary Runs (Chess)



Local Optima Through Wind Discretization



Rotor Size of the Affected Turbine



Example of a small wind farm

- Enercon E101 with 3 MW at 12 m/s
- Calculating with 40% full load hours
  → average 1.2 MW
- Small wind farm with 10 turbines
  → more than 100 GWh/a
- With price at 0.10 €/kWh
  → 10 M€/a
- Even small improvements lead to large values

CMA-ES

- By N. Hansen
- Newest Change
  - 14/12/06: meta_parameters now only as annotations in ## comments
- Python implementation

```
self.__sigma = 10: corresponds to 10m

res = cma.fmin(
            objective_function = fitness_function,
            x0 = XY,
            sigma0 = self.__sigma,
            options = {'maxfevals': evaluations,
                       'verb_log': True,
                       'verb_filenameprefix': file_prefix},
            args = [self.__solution]
)
```

Wind Model

- Model based on wind distributions by Kusiak and Song:

$$E(\mathbf{t}_i) = \int_0^{360} p_\theta(\mathbf{t}_i, \theta) \cdot E_\theta(\mathbf{t}_i, \theta) d\theta$$

$$E_\theta(\mathbf{t}_i, \theta) = \int_0^{\infty} \beta_i(v) \cdot p_v(v, k(\mathbf{t}_i, \theta), c(\mathbf{t}_i, \theta)) dv$$

Wind Model

- Model based on wind distributions by Kusiak and Song:

$$E(\mathbf{t}_i) = \int_0^{360} p_\theta(\mathbf{t}_i, \theta) \cdot E_\theta(\mathbf{t}_i, \theta) d\theta$$

$$E_\theta(\mathbf{t}_i, \theta) = \int_0^{\infty} \beta_i(v) \cdot p_v(v, k(\mathbf{t}_i, \theta), c(\mathbf{t}_i, \theta)) dv$$

Wind Model

- Model based on wind distributions by Kusiak and Song:

$$E(\mathbf{t}_i) = \int_0^{360} p_\theta(\mathbf{t}_i, \theta) \cdot E_\theta(\mathbf{t}_i, \theta) d\theta$$

$$E_\theta(\mathbf{t}_i, \theta) = \int_0^{\infty} \beta_i(v) \cdot p_v(v, k(\mathbf{t}_i, \theta), c(\mathbf{t}_i, \theta)) dv$$

Wind Model

- Model based on wind distributions by Kusiak and Song:

$$E(\mathbf{t}_i) = \int_0^{360} p_\theta(\mathbf{t}_i, \theta) \cdot E_\theta(\mathbf{t}_i, \theta) d\theta$$

$$E_\theta(\mathbf{t}_i, \theta) = \int_0^{\infty} \beta_i(v) \cdot \boxed{p_v(v, k(\mathbf{t}_i, \theta), c(\mathbf{t}_i, \theta)) dv}$$

- ***Weibull distribution: Wind speed distribution from the wind roses***

Wind Model

- Model based on wind distributions by Kusiak and Song:

$$E(\mathbf{t}_i) = \int_0^{360} \boxed{p_\theta(\mathbf{t}_i, \theta)} \cdot E_\theta(\mathbf{t}_i, \theta) d\theta$$

$$E_\theta(\mathbf{t}_i, \theta) = \int_0^{\infty} \beta_i(v) \cdot p_v(v, k(\mathbf{t}_i, \theta), c(\mathbf{t}_i, \theta)) dv$$

- Wind speed distribution from the wind roses
- ***Wind direction distribution from the wind roses***

Wind Model

- Model based on wind distributions by Kusiak and Song:

$$E(\mathbf{t}_i) = \int_0^{360} p_\theta(\mathbf{t}_i, \theta) \cdot E_\theta(\mathbf{t}_i, \theta) d\theta$$

$$E_\theta(\mathbf{t}_i, \theta) = \int_0^{\infty} \boxed{\beta_i(v)} \cdot p_v(v, k(\mathbf{t}_i, \theta), c(\mathbf{t}_i, \theta)) dv$$

- Wind speed distribution from the wind roses
- Wind direction distribution from the wind roses
- ***Turbine power curve***