

Metody optymalizacji – wprowadzenie

dr hab. inż. Maciej Komosiński, prof. PP

Instytut Informatyki
Politechnika Poznańska
www.cs.put.poznan.pl/mkomosinski

Literatura

- D.E. Goldberg. Algorytmy genetyczne i zastosowania, WNT, 2003
- Z. Michalewicz. Algorytmy genetyczne + struktury danych = programy ewolucyjne, WNT Warszawa, 2003
- E. Aarts, J. Korst. Simulated Annealing and Boltzmann Machines – A stochastic approach to combinatorial optimization and neural computing, Willey, 1988
- F. Glover, M. Laguna. Tabu search, Kluwer academic publishers, Boston, 1997
- Z. Michalewicz, D.B. Fogel. How to Solve It: Modern Heuristics, Springer, 2000

Optymalizacja?

- Sposoby znajdowania dobrych (najlepszych) rozwiązań
- W trudnych problemach

Optymalizacja?

- Sposoby znajdowania dobrych (najlepszych) rozwiązań
- W trudnych problemach
- bez wymogu bycia ekspertem w dziedzinach, z których pochodzą te problemy

Co to znaczy „problem optymalizacyjny”?

- Problem istnieje, gdy zauważono różnicę między stanem zastanym, a pożądanym.

Co to znaczy „problem optymalizacyjny”?

- Problem istnieje, gdy zauważono różnicę między stanem zastanym, a pożądanym.
- Rozwiązanie problemu polega na działaniu w celu zmniejszenia różnicy między stanem zastanym, a pożądanym.

Co to znaczy „problem optymalizacyjny”?

- Problem istnieje, gdy zauważono różnicę między stanem zastanym, a pożądanym.
- Rozwiązanie problemu polega na działaniu w celu zmniejszenia różnicy między stanem zastanym, a pożądanym.
- Często osiągalne jest znaczne polepszenie – jakość, zasoby (czas, pieniądze).

Co to znaczy „problem optymalizacyjny”?

- Problem istnieje, gdy zauważono różnicę między stanem zastanym, a pożądanym.
- Rozwiązanie problemu polega na działaniu w celu zmniejszenia różnicy między stanem zastanym, a pożądanym.
- Często osiągalne jest znaczne polepszenie – jakość, zasoby (czas, pieniądze).
- Umiejętność optymalizacji – *know-how*

Co to znaczy „problem optymalizacyjny”?

- Problem istnieje, gdy zauważono różnicę między stanem zastanym, a pożądanym.
- Rozwiązanie problemu polega na działaniu w celu zmniejszenia różnicy między stanem zastanym, a pożądanym.
- Często osiągalne jest znaczne polepszenie – jakość, zasoby (czas, pieniądze).
- Umiejętność optymalizacji – *know-how*
- Wykorzystywana od dawna... <https://www.youtube.com/watch?v=3QQjaZ1FLHE>

Terminologia i definicje

- Optymalizacja a wspomaganie decyzji

Terminologia i definicje

- Optymalizacja a wspomaganie decyzji
- Problem, instancja problemu, przestrzeń rozwiązań, kryterium oceny, funkcja celu, algorytm

Trzy składowe problemu optymalizacji

- rozwiązanie (wariant decyzyjny): $x = [x_1, x_2, \dots, x_n]$
- kryteria oceny rozwiązania i funkcja celu: $z = f(x)$
- ograniczenia

What are *combinatorial* optimization problems?

(as opposed to *numerical* optimization problems)

- Many real world problems can be expressed as a combination of elements
 - e.g. Traveling Salesman Problem is a permutation of integers

What are *combinatorial* optimization problems?

(as opposed to *numerical* optimization problems)

- Many real world problems can be expressed as a combination of elements
 - e.g. Traveling Salesman Problem is a permutation of integers
- More formally:
 - minimizing (or maximizing)
 - a function $f(x)$
 - subject to constraints, e.g. $g(x) > 0$
 - x is a solution, often encoded with vectors of symbols

What are *combinatorial* optimization problems?

(as opposed to *numerical* optimization problems)

- Many real world problems can be expressed as a combination of elements
 - e.g. Traveling Salesman Problem is a permutation of integers
- More formally:
 - minimizing (or maximizing)
 - a function $f(x)$
 - subject to constraints, e.g. $g(x) > 0$
 - x is a solution, often encoded with vectors of symbols
- NP-hard!

Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa

Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa

Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa
 - Wykres: liczba_rozwiązań(n)

Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa
 - Wykres: liczba_rozwiązań(n)
 - Wykres: złożoność_czasowa(n)

Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa
 - Wykres: liczba_rozwiązań(n)
 - Wykres: złożoność_czasowa(n)
 - Optymalizacja a łamanie szyfrów

Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa
 - Wykres: liczba_rozwiązań(n)
 - Wykres: złożoność_czasowa(n)
 - Optymalizacja a łamanie szyfrów
 - Maszyny Turinga: DTM („tradycyjny komputer”), NDTM („zgadywacz, który zawsze dobrze zgaduje”)

Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa
 - Wykres: liczba_rozwiązań(n)
 - Wykres: złożoność_czasowa(n)
 - Optymalizacja a łamanie szyfrów
 - Maszyny Turinga: DTM („tradycyjny komputer”), NDTM („zgadywacz, który zawsze dobrze zgaduje”)
 - Transformacja wielomianowa, pseudowielomianowa

Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa
 - Wykres: liczba_rozwiązań(n)
 - Wykres: złożoność_czasowa(n)
 - Optymalizacja a łamanie szyfrów
 - Maszyny Turinga: DTM („tradycyjny komputer”), NDTM („zgadywacz, który zawsze dobrze zgaduje”)
 - Transformacja wielomianowa, pseudowielomianowa
 - Problemy optymalizacyjne („znajdź najlepsze rozwiązanie”) a problemy decyzyjne („czy istnieje rozwiązanie takie, że...”)

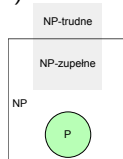
Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa
 - Wykres: liczba_rozwiązań(n)
 - Wykres: złożoność_czasowa(n)
 - Optymalizacja a łamanie szyfrów
 - Maszyny Turinga: DTM („tradycyjny komputer”), NDTM („zgadywacz, który zawsze dobrze zgaduje”)
 - Transformacja wielomianowa, pseudowielomianowa
 - Problemy optymalizacyjne („znajdź najlepsze rozwiązanie”) a problemy decyzyjne („czy istnieje rozwiązanie takie, że...”)
 - Klasy złożoności problemów:

P (łatwo znaleźć, łatwo zweryfikować), NP (łatwo zweryfikować), NP-C, strongly NP-C, NP-H, EXP, R, ...

youtube: [\[skrót\]](#), [\[konkretne, zwięzłe wprowadzenie\]](#), [\[nieformalne, gawędziarskie wprowadzenie\]](#), [\[przykłady szacowania złożoności czasowej prostych algorytmów \(z dokarmianiem\)\]](#),

pełniejsze wyjaśnienie [\[1\]](#), [\[2\]](#), [\[3\]](#)



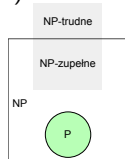
Złożoność obliczeniowa

- Złożoność czasowa i pamięciowa
 - Przykłady: porównanie/konfrontacja par, wybór podzbioru cech, najkrótsza trasa
 - Wykres: liczba_rozwiązań(n)
 - Wykres: złożoność_czasowa(n)
 - Optymalizacja a łamanie szyfrów
 - Maszyny Turinga: DTM („tradycyjny komputer”), NDTM („zgadywacz, który zawsze dobrze zgaduje”)
 - Transformacja wielomianowa, pseudowielomianowa
 - Problemy optymalizacyjne („znajdź najlepsze rozwiązanie”) a problemy decyzyjne („czy istnieje rozwiązanie takie, że...”)
 - Klasy złożoności problemów:

P (łatwo znaleźć, łatwo zweryfikować), NP (łatwo zweryfikować), NP-C, strongly NP-C, NP-H, EXP, R, ...

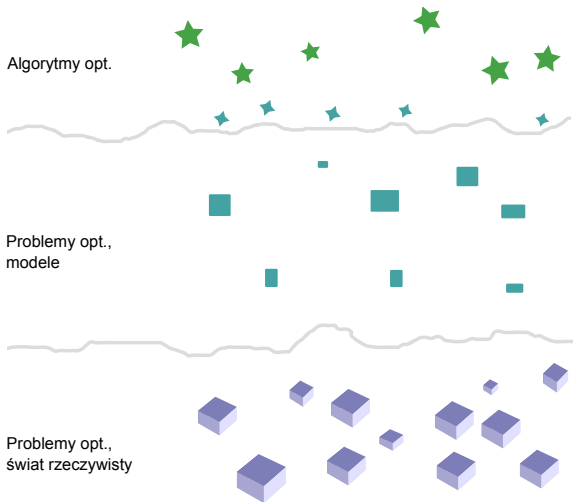
youtube: [\[skrót\]](#), [\[konkretne, związane wprowadzenie\]](#), [\[nieformalne, gawędziarskie wprowadzenie\]](#), [\[przykłady szacowania złożoności czasowej prostych algorytmów \(z dokarmianiem\)\]](#),

pełniejsze wyjaśnienie [\[1\]](#), [\[2\]](#), [\[3\]](#)

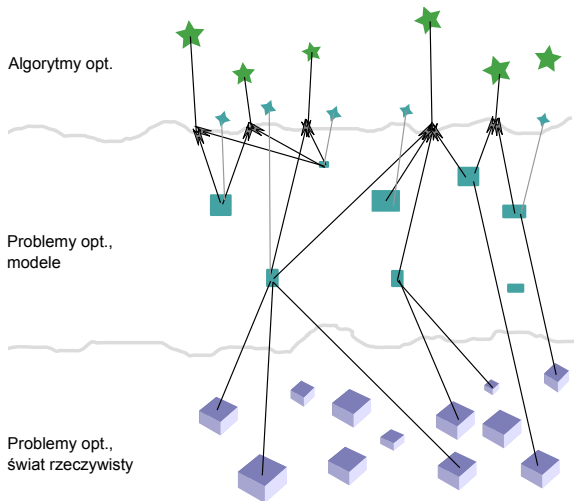


- Najprostsze algorytmy: heurystyka, losowy, wyczerpujący-dokładny

Problemy, modele, instancje, algorytmy



Problemy, modele, instancje, algorytmy – relacje



Motivations for developing optimization methods

- Economy
- Machine vision
- Molecular biology
- Optimal power flow
- Structural optimization
- Robotics
- Database systems
- Computer graphics

- Medicine
- Telecommunications
- Artificial intelligence
- Integrated circuit design
automation
- Computer architecture design
- Computer networks
- Image processing
- Security

Motivations for developing optimization methods

- Economy
- Machine vision
- Molecular biology
- Optimal power flow
- Structural optimization
- Robotics
- Database systems
- Computer graphics

- Medicine
- Telecommunications
- Artificial intelligence
- Integrated circuit design automation
- Computer architecture design
- Computer networks
- Image processing
- Security

NP-hard.

Sample applications of optimization methods

- Design of electronic circuits (VLSI)
- Telecommunication network design
- Knowledge discovery / Machine Learning
- Neural network training & design
- Automatic control
- Business scheduling and planning
- Games
- Self-adapting computer programs
- Test-data generation
- Medical image analysis
- DNA Sequencing

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji
- Zdecyduj gdzie w tekście umieścić rysunki

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji
- Zdecyduj gdzie w tekście umieścić rysunki
- Ustal położenie układów scalonych na płycie drukowanej

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji
- Zdecyduj gdzie w tekście umieścić rysunki
- Ustal położenie układów scalonych na płycie drukowanej
- Wybierz część działów/pracowników biura do przeniesienia do innej lokalizacji

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwozu towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji
- Zdecyduj gdzie w tekście umieścić rysunki
- Ustal położenie układów scalonych na płycie drukowanej
- Wybierz część działów/pracowników biura do przeniesienia do innej lokalizacji
- Wybierz najlepszy przebieg linii metra

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji
- Zdecyduj gdzie w tekście umieścić rysunki
- Ustal położenie układów scalonych na płycie drukowanej
- Wybierz część działów/pracowników biura do przeniesienia do innej lokalizacji
- Wybierz najlepszy przebieg linii metra
- Rozmieść w najkorzystniejszych miejscach przystanki autobusowe

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji
- Zdecyduj gdzie w tekście umieścić rysunki
- Ustal położenie układów scalonych na płycie drukowanej
- Wybierz część działów/pracowników biura do przeniesienia do innej lokalizacji
- Wybierz najlepszy przebieg linii metra
- Rozmieść w najkorzystniejszych miejscach przystanki autobusowe
- Podaj najlepszą kolejność składania samochodów na taśmie produkcyjnej

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji
- Zdecyduj gdzie w tekście umieścić rysunki
- Ustal położenie układów scalonych na płycie drukowanej
- Wybierz część działów/pracowników biura do przeniesienia do innej lokalizacji
- Wybierz najlepszy przebieg linii metra
- Rozmieść w najkorzystniejszych miejscach przystanki autobusowe
- Podaj najlepszą kolejność składania samochodów na taśmie produkcyjnej
- Podaj harmonogram wyłączania i konserwacji elektrowni

Praktyczne przykłady problemów NP-trudnych (1/7)

- Połącz ludzi w zespoły wedle kompetencji
- Rozmieść biura w budynku firmy
- Ustal trasę dla śmieciarki (lub rozwoju towarów z marketu)
- Narysuj najbardziej czytelny schemat/graf na podstawie specyfikacji
- Zdecyduj gdzie w tekście umieścić rysunki
- Ustal położenie układów scalonych na płycie drukowanej
- Wybierz część działów/pracowników biura do przeniesienia do innej lokalizacji
- Wybierz najlepszy przebieg linii metra
- Rozmieść w najkorzystniejszych miejscach przystanki autobusowe
- Podaj najlepszą kolejność składania samochodów na taśmie produkcyjnej
- Podaj harmonogram wyłączania i konserwacji elektrowni
- Znajdź identyczne fragmenty w sekwencjach DNA

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu
- Podaj wartości zmiennych, dla których wyrażenie logiczne jest prawdą

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu
- Podaj wartości zmiennych, dla których wyrażenie logiczne jest prawdą
- Określ kolejność i czas nadawania reklam w radiu/TV

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu
- Podaj wartości zmiennych, dla których wyrażenie logiczne jest prawdą
- Określ kolejność i czas nadawania reklam w radiu/TV
- Poprowadź sieć tak, by najtaniej połączyć budynki

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu
- Podaj wartości zmiennych, dla których wyrażenie logiczne jest prawdą
- Określ kolejność i czas nadawania reklam w radiu/TV
- Poprowadź sieć tak, by najtaniej połączyć budynki
- Wybierz akcje, w które zainwestujesz posiadane środki

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu
- Podaj wartości zmiennych, dla których wyrażenie logiczne jest prawdą
- Określ kolejność i czas nadawania reklam w radiu/TV
- Poprowadź sieć tak, by najtaniej połączyć budynki
- Wybierz akcje, w które zainwestujesz posiadane środki
- Wybierz pliki do nagrania/archiwizacji na nośnikach danych o określonej pojemności

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu
- Podaj wartości zmiennych, dla których wyrażenie logiczne jest prawdą
- Określ kolejność i czas nadawania reklam w radiu/TV
- Poprowadź sieć tak, by najtaniej połączyć budynki
- Wybierz akcje, w które zainwestujesz posiadane środki
- Wybierz pliki do nagrania/archiwizacji na nośnikach danych o określonej pojemności
- Przydziel pracę pracownikom uwzględniając ograniczenia

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu
- Podaj wartości zmiennych, dla których wyrażenie logiczne jest prawdą
- Określ kolejność i czas nadawania reklam w radiu/TV
- Poprowadź sieć tak, by najtaniej połączyć budynki
- Wybierz akcje, w które zainwestujesz posiadane środki
- Wybierz pliki do nagrania/archiwizacji na nośnikach danych o określonej pojemności
- Przydziel pracę pracownikom uwzględniając ograniczenia
- Wybierz cechy najbardziej odróżniające dwie grupy osób

Praktyczne przykłady problemów NP-trudnych (2/7)

- Zaprojektuj najwygodniejszy układ klawiszy na klawiaturze
- Przydziel zadania do procesorów
- Przypisz oddziały szpitala do posiadanych lokalizacji
- Ułóż plan zajęć
- Zaprojektuj najlepszą antenę lub skrzydło samolotu
- Podaj wartości zmiennych, dla których wyrażenie logiczne jest prawdą
- Określ kolejność i czas nadawania reklam w radiu/TV
- Poprowadź sieć tak, by najtaniej połączyć budynki
- Wybierz akcje, w które zainwestujesz posiadane środki
- Wybierz pliki do nagrania/archiwizacji na nośnikach danych o określonej pojemności
- Przydziel pracę pracownikom uwzględniając ograniczenia
- Wybierz cechy najbardziej odróżniające dwie grupy osób
- Zaproponuj najbardziej zwięzłą hierarchię klas (sposób dziedziczenia)

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płytce drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku
- Stwórz program grający w szachy

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku
- Stwórz program grający w szachy
- Wykryj minimalny zestaw różnic w dwóch tekstach

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku
- Stwórz program grający w szachy
- Wykryj minimalny zestaw różnic w dwóch tekstach
- Uprość wyrażenie logiczne lub algebraiczne

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płytce drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku
- Stwórz program grający w szachy
- Wykryj minimalny zestaw różnic w dwóch tekstach
- Uprość wyrażenie logiczne lub algebraiczne
- Rekrutacja: przydziel kandydatów do kierunków uwzględniając ich preferencje

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku
- Stwórz program grający w szachy
- Wykryj minimalny zestaw różnic w dwóch tekstach
- Uprość wyrażenie logiczne lub algebraiczne
- Rekrutacja: przydziel kandydatów do kierunków uwzględniając ich preferencje
- Dobierz wagi w sieci neuronowej (naucz ją)

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku
- Stwórz program grający w szachy
- Wykryj minimalny zestaw różnic w dwóch tekstach
- Uprość wyrażenie logiczne lub algebraiczne
- Rekrutacja: przydziel kandydatów do kierunków uwzględniając ich preferencje
- Dobierz wagi w sieci neuronowej (naucz ją)
- Podaj dowód twierdzenia

Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku
- Stwórz program grający w szachy
- Wykryj minimalny zestaw różnic w dwóch tekstach
- Uprość wyrażenie logiczne lub algebraiczne
- Rekrutacja: przydziel kandydatów do kierunków uwzględniając ich preferencje
- Dobierz wagi w sieci neuronowej (naucz ją)
- Podaj dowód twierdzenia
- Stwórz pożądane wyrażenie algebraiczne albo program

[▶ przykład](#)

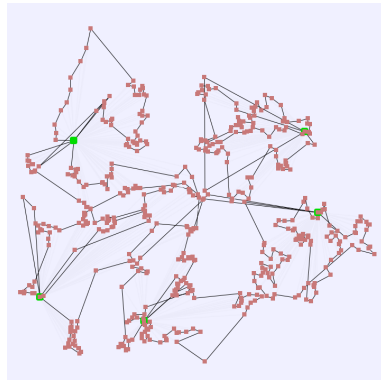
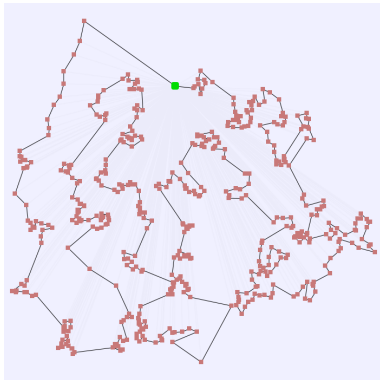
Praktyczne przykłady problemów NP-trudnych (3/7)

- „Serwis randkowy”: ustal pary wśród użytkowników
- Ustal najlepszą kolejność wiercenia otworków w płycie drukowanej
- Określ trajektorie lotu satelitów (pokrycie zdjęciami obszarów Ziemi)
- Zdecyduj ile i jakie rodzaje zbóż uprawiać w danym roku
- Stwórz program grający w szachy
- Wykryj minimalny zestaw różnic w dwóch tekstach
- Uprość wyrażenie logiczne lub algebraiczne
- Rekrutacja: przydziel kandydatów do kierunków uwzględniając ich preferencje
- Dobierz wagi w sieci neuronowej (naucz ją)
- Podaj dowód twierdzenia
- Stwórz pożądane wyrażenie algebraiczne albo program ▶ przykład
- Wybierz najszybszy sposób realizacji zapytania SQL do bazy danych

NP-trudność w praktyce i na własnym podwórku (4/7)

OptiFacility, <http://optifacility.mooncoder.com/>

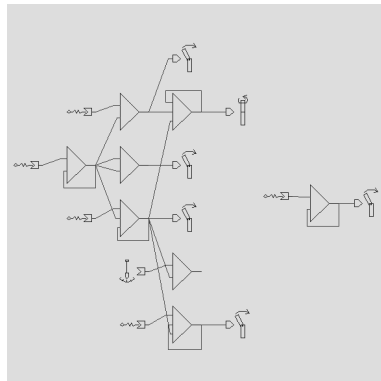
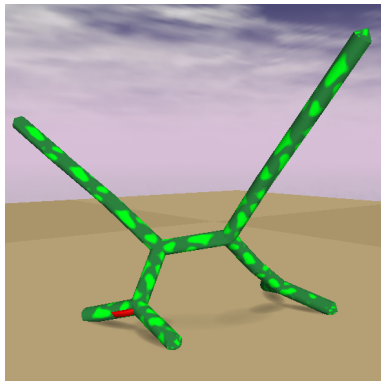
Logistyka, transport, dostawcy, trasy...



NP-trudność w praktyce i na własnym podwórku (5/7)

Framsticks, <http://www.framsticks.com/>

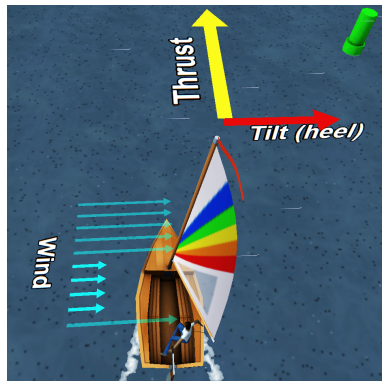
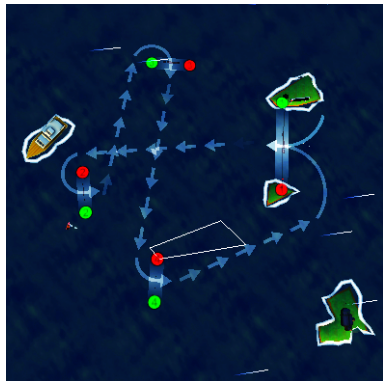
Konstrukcje, systemy sterujące, sieci neuronowe...



NP-trudność w praktyce i na własnym podwórku (6/7)

Top Sailor, <http://sailor.mooncoder.com/>

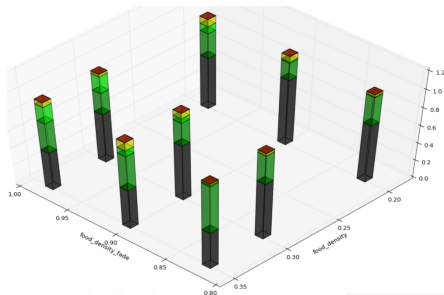
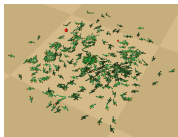
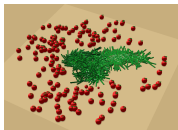
Planowanie tras, sterowanie, strategia...



NP-trudność w praktyce i na własnym podwórku (7/7)

Framsticks, <http://www.framsticks.com/>

Kiedy powstaje... życie? Wykres 2D z 11D



Zadanie domowe

Wybierz kilka z wymienionych na slajdach 1/7, 2/7 i 3/7 problemów i zapisz czym jest zbiór rozwiązań, jaki jest duży i jak ograniczony, jak wymienić jego wszystkie elementy, oraz jakie są kryteria oceny i jak je automatycznie obliczyć dla każdego możliwego rozwiązania.

Dlaczego problemy mogą być trudne do rozwiązania?

- 1 Problem jest bardzo złożony
 - użycie modeli uproszczonych i rezultaty są bezużyteczne

Dlaczego problemy mogą być trudne do rozwiązania?

- ❶ Problem jest bardzo złożony
 - użycie modeli uproszczonych i rezultaty są bezużyteczne
- ❷ Duża liczba możliwych rozwiązań
 - przeszukanie całej przestrzeni rozwiązań dopuszczalnych w celu znalezienia najlepszego jest nierealne

Dlaczego problemy mogą być trudne do rozwiązania?

- ❶ Problem jest bardzo złożony
 - użycie modeli uproszczonych i rezultaty są bezużyteczne
- ❷ Duża liczba możliwych rozwiązań
 - przeszukanie całej przestrzeni rozwiązań dopuszczalnych w celu znalezienia najlepszego jest nierealne
- ❸ Funkcja oceny jest obarczona niepewnością

Dlaczego problemy mogą być trudne do rozwiązania?

- ❶ Problem jest bardzo złożony
 - użycie modeli uproszczonych i rezultaty są bezużyteczne
- ❷ Duża liczba możliwych rozwiązań
 - przeszukanie całej przestrzeni rozwiązań dopuszczalnych w celu znalezienia najlepszego jest nierealne
- ❸ Funkcja oceny jest obarczona niepewnością
- ❹ Potencjalne rozwiązania mocno ograniczone
 - już znalezienie jednego dopuszczalnego jest problemem

Trudność #1. Modelowanie problemu

PROBLEM $\leftrightarrow$ MODEL $\leftrightarrow$ ROZWIĄZANIE

- Model to przybliżenie rzeczywistości
- Rozwiązanie $\rightarrow$ problem
- Np. problem transportowy z nieliniową i nieciągłą funkcją celu, lub badania operacyjne
- Jeśli problem rzeczywisty nie jest identyczny z istniejącym, klasycznym modelem...
 - uprościć problem, żeby pasował do istniejącego modelu i metody rozwiązania
 - wykorzystać nietradycyjne podejście

Trudność #2. Duża liczba rozwiązań

Np. problem spełnienia wyrażenia logicznego (SAT)

$$F(x) = (x_{13} \vee \bar{x}_{23} \vee x_{34}) \wedge (\bar{x}_{13} \vee x_{23} \vee \bar{x}_{34}) \wedge \dots = \text{TRUE}$$

Dla 100 zmiennych (dwie możliwości dla zmiennej)

$$|S| =$$

Trudność #2. Duża liczba rozwiązań

Np. problem spełnienia wyrażenia logicznego (SAT)

$$F(x) = (x_{13} \vee \bar{x}_{23} \vee x_{34}) \wedge (\bar{x}_{13} \vee x_{23} \vee \bar{x}_{34}) \wedge \dots = \text{TRUE}$$

Dla 100 zmiennych (dwie możliwości dla zmiennej)

$$|S| = 2^{100} \approx 10^{30}$$

Trudność #2. Duża liczba rozwiązań

Np. problem spełnienia wyrażenia logicznego (SAT)

$$F(x) = (x_{13} \vee \bar{x}_{23} \vee x_{34}) \wedge (\bar{x}_{13} \vee x_{23} \vee \bar{x}_{34}) \wedge \dots = \text{TRUE}$$

Dla 100 zmiennych (dwie możliwości dla zmiennej)

$$|S| = 2^{100} \approx 10^{30}$$

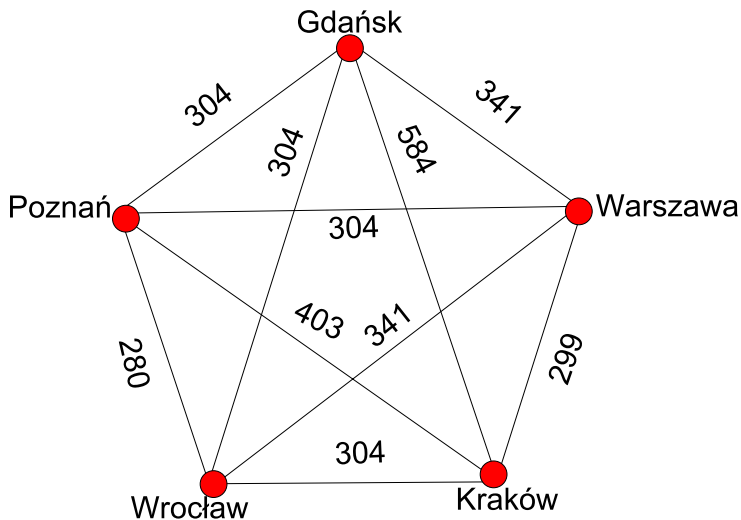
czasem możliwa redukcja przestrzeni (przykład ROADEF)

Problemy optymalizacji kombinatorycznej

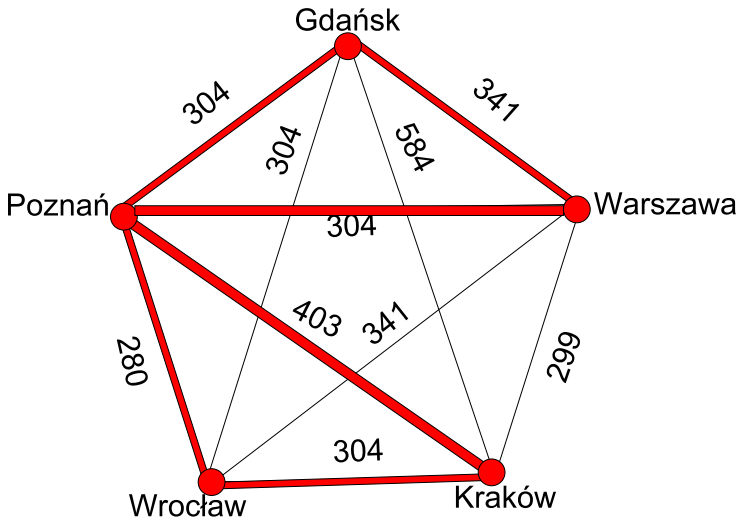
Np. permutacja obiektów jako rozwiązanie:

- N – zbiór numerów obiektów $\{1, \dots, n\}$
- $\Pi(i)$ – numer obiektu na pozycji i
- $\Pi = \Pi(1), \Pi(2), \dots, \Pi(n)$
- Cel: znaleźć permutację z optymalną wartością funkcji celu

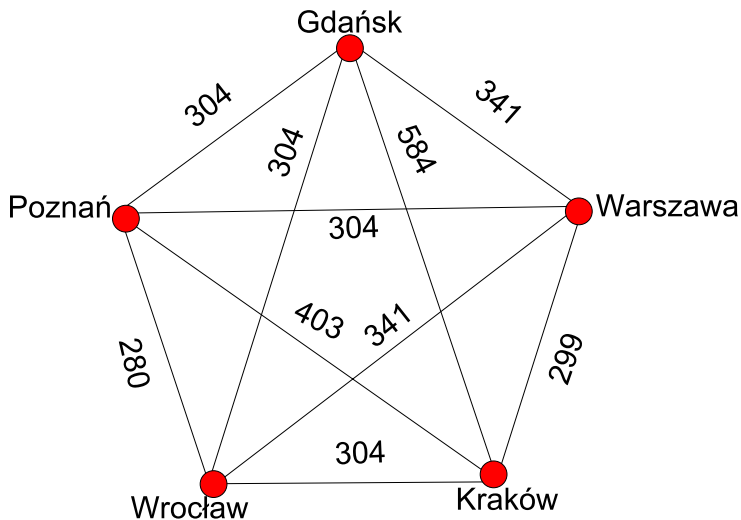
Problem komiwojażera (TSP)



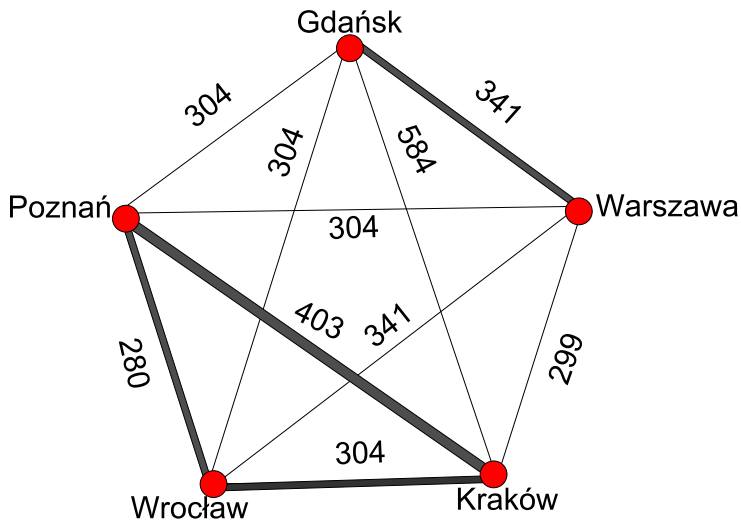
Problem komiwojażera (TSP)



Problem komiwojażera (TSP)



Problem komiwojażera (TSP)



Problem komiwojażera (TSP)

$$|S| = \frac{n!}{2n} = \frac{(n-1)!}{2}$$

Każda trasa wyrażona na $2n$ różnych sposobów,
 $n!$ sposobów permutacji n liczb

10 miast	181 440
20 miast	60 822 550 204 416 000
50 miast	304 251 932 017 133 780 436 126 081 660 647 688 443 776 415 689 605 120 000 000 000

Problem komiwojażera (TSP)

$$|S| = \frac{n!}{2n} = \frac{(n-1)!}{2}$$

Każda trasa wyrażona na $2n$ różnych sposobów,
 $n!$ sposobów permutacji n liczb

10 miast	181 440
20 miast	60 822 550 204 416 000
50 miast	304 251 932 017 133 780 436 126 081 660 647 688 443 776 415 689 605 120 000 000 000

	n=10	1 024
Dla porównania: wybór podzbioru	n=20	1 048 576
	n=50	1 125 899 906 842 624

Problem komiwojażera (TSP)

$$|S| = \frac{n!}{2n} = \frac{(n-1)!}{2}$$

Każda trasa wyrażona na $2n$ różnych sposobów,
 $n!$ sposobów permutacji n liczb

10 miast	181 440
20 miast	60 822 550 204 416 000
50 miast	304 251 932 017 133 780 436 126 081 660 647 688 443 776 415 689 605 120 000 000 000

	n=10	1 024
Dla porównania: wybór podzbioru	n=20	1 048 576
	n=50	1 125 899 906 842 624

	n=10	45
Dla porównania: ocena par	n=20	190
	n=50	1 225

Problem kwadratowego przydziału (QAP)

- Dane
 - Odległości pomiędzy możliwymi lokalizacjami
 - Przepływy pomiędzy czynnościami
- Przykład – lokalizacja personelu medycznego

Problem podziału grafu (GPP)

Dane:

- Graf $G(V, E)$ składający się z n wierzchołków
- $V = \{v_1, v_2, \dots, v_n\}$
- zbiór nieorientowanych łuków E łączących pary wierzchołków

Problem podziału grafu (GPP)

Dane:

- Graf $G(V, E)$ składający się z n wierzchołków
- $V = \{v_1, v_2, \dots, v_n\}$
- zbiór nieorientowanych łuków E łączących pary wierzchołków
 - E_{ij} – macierz połączeń
 - $E_{ij} = 1$ jeśli v_i jest połączony z v_j , 0 w przeciwnym wypadku
 - $E_{ij} = E_{ji}$

Problem podziału grafu (GPP)

Dane:

- Graf $G(V, E)$ składający się z n wierzchołków
- $V = \{v_1, v_2, \dots, v_n\}$
- zbiór nieorientowanych łuków E łączących pary wierzchołków
 - E_{ij} – macierz połączeń
 - $E_{ij} = 1$ jeśli v_i jest połączony z v_j , 0 w przeciwnym wypadku
 - $E_{ij} = E_{ji}$
- Należy dokonać podziału grafu G na (dwa) równoliczne rozłączne podzbiory V_1 i V_2 takie, że $V_1 \cup V_2 = V$

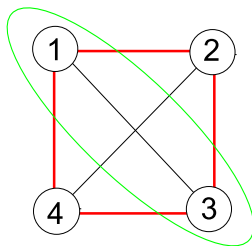
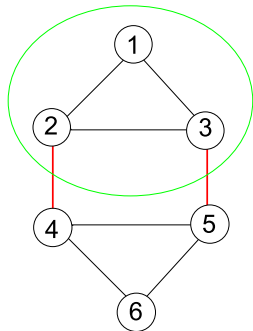
Problem podziału grafu (GPP)

Dane:

- Graf $G(V, E)$ składający się z n wierzchołków
- $V = \{v_1, v_2, \dots, v_n\}$
- zbiór nieorientowanych łuków E łączących pary wierzchołków
 - E_{ij} – macierz połączeń
 - $E_{ij} = 1$ jeśli v_i jest połączony z v_j , 0 w przeciwnym wypadku
 - $E_{ij} = E_{ji}$
- Należy dokonać podziału grafu G na (dwa) równoliczne rozłączne podzbiory V_1 i V_2 takie, że $V_1 \cup V_2 = V$
- Minimalizowana liczba połączeń pomiędzy podzbiorami wynosi

$$c(V_1, V_2) = \sum_{i \in V_1, j \in V_2} E_{ij}$$

Problem bisekcji grafu (2-GPP)



Zastosowania:

VLSI design, networks design, data mining, geographical information systems, job scheduling

Problem plecakowy

- Dany jest zbiór I elementów
- Każdy element $i = 1, \dots, I$ ma wagę w_i i wartość c_i
- W plecaku o pojemności W należy umieścić elementy o jak największej łącznej wartości i łącznej wadze nie przekraczającej W
- Przykładowa interpretacja – wybór projektów inwestycyjnych
 - Elementy – projekty inwestycyjne
 - Wagi – koszty projektów
 - Wartości – zyski z projektów

Common Problems

- General function optimization
- Traveling Salesman Problem (TSP)
- Quadratic Assignment Problem (QAP)
- Graph Coloring & Partitioning (GPP)
- Minimum Spanning Tree Problem (MSTP)
- Vehicle Routing (VRP)
- Single & Multiple Knapsack
- Set Partitioning (SPP) & Set Covering Problems (SCP)
- Cutting stock problem (CSTP)
- 2-Dimensional Packing Problem (2PP)
- Processor Allocation Problem
- Staff Scheduling Problems
- Job Shop & Project Scheduling (PSP)

Compendium of NP optimization problems

<https://www.nada.kth.se/~viggo/problemlist/compendium.html>

- Graph Theory
 - Covering and Partitioning
 - Subgraphs and Supergraphs
 - Vertex Ordering
 - Iso- and Other Morphisms
- Network Design
 - Spanning Trees
 - Cuts and Connectivity
 - Routing Problems
 - Flow Problems
- Sets and Partitions
 - Covering, Hitting and Splitting
 - Weighted Set Problems
- Storage and Retrieval
 - Data Storage
 - Compression and Representation
- Sequencing and Scheduling
 - Sequencing on One Processor
 - Multiprocessor Scheduling
 - Shop Scheduling
- Mathematical Programming
- Algebra and Number Theory
 - Solvability of Equations
- Games and Puzzles
- Logic
 - Propositional Logic
- Automata and Language Theory
 - Automata Theory
 - Formal Languages
- Program Optimization
 - Code Generation

Compendium of NP optimization problems

<https://www.nada.kth.se/~viggo/problemlist/compendium.html>

- Graph Theory
 - Covering and Partitioning
 - Subgraphs and Supergraphs
 - Vertex Ordering
 - Iso- and Other Morphisms
- **Network Design**
 - **Spanning Trees**
 - **Cuts and Connectivity**
 - **Routing Problems**
 - **Flow Problems**
- Sets and Partitions
 - Covering, Hitting and Splitting
 - Weighted Set Problems
- Storage and Retrieval
 - Data Storage
 - Compression and Representation
- Sequencing and Scheduling
 - Sequencing on One Processor
 - Multiprocessor Scheduling
 - Shop Scheduling
- Mathematical Programming
- Algebra and Number Theory
 - Solvability of Equations
- Games and Puzzles
- Logic
 - Propositional Logic
- Automata and Language Theory
 - Automata Theory
 - Formal Languages
- Program Optimization
 - Code Generation

Network design NP problems

• Spanning Trees

- MIN K-SPANNING TREE
- MIN DEGREE SPANNING TREE
- MIN GEOMETRIC 3-DEGREE SPANNING TREE
- MAX LEAF SPANNING TREE
- MAX MIN METRIC K-SPANNING TREE
- MIN DIAMETER SPANNING SUBGRAPH
- MIN COMMUNICATION COST SPANNING TREE
- MIN STEINER TREE
- MIN GEOMETRIC STEINER TREE
- MIN GENERALIZED STEINER NET
- MIN ROUTING TREE CONGESTION
- MAX MIN SPANNING TREE DELETING K EDGES
- MIN UPGRADING SPANNING TREE

• Routing Problems

- MIN TRAVELING SALESPERSON
- MIN METRIC TRAVELING SALESPERSON PROBLEM
- MIN GEOMETRIC TRAVELING SALESPERSON
- MIN METRIC TRAVELING K-SALESPERSON PROBLEM
- MIN METRIC BOTTLENECK WANDERING SALESPERSON PROBLEM
- MIN CHINESE POSTMAN FOR MIXED GRAPHS
- MIN K-CHINESE POSTMAN PROBLEM
- MIN STACKER CRANE PROBLEM
- MIN K-STACKER CRANE PROBLEM
- MIN GENERAL ROUTING
- LONGEST PATH
- SHORTEST WEIGHT-CONSTRAINED PATH
- MIN RECTILINEAR GLOBAL ROUTING
- MIN TRAVELING REPAIRMAN
- MAX QUADRATIC ASSIGN

• Cuts and Connectivity

- MAX CUT
- MIN CROSSING NUMBER
- MAX DIRECTED CUT
- MAX K-CUT
- MIN NET INHIBITION ON PLANAR GRAPHS
- MIN K-CUT
- MIN VERTEX K-CUT
- MIN MULTIWAY CUT
- MIN MULTI-CUT
- MIN RATIO-CUT
- MIN B-BALANCED CUT
- MIN B-VERTEX SEPARATOR
- MIN QUOTIENT CUT
- MIN K-VERTEX CONNECTED SUBGRAPH
- MIN K-EDGE CONNECTED SUBGRAPH
- MIN BICONNECTIVITY AUGMENTATION
- MIN STRONG CONNECTIVITY AUGMENTATION
- MIN BOUNDED DIAMETER AUGMENTATION

• Flow Problems

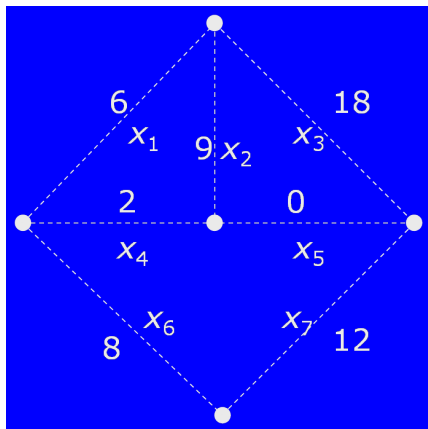
- MAX PRIORITY FLOW
- MAX INTEGRAL K-MULTICOMMODITY FLOW ON TREE
- MAX DISJOINT CONNECTING PATHS
- MIN MAX DISJOINT CONNECTING PATHS
- MIN SINGLE-SINK EDGE INSTALLATION
- MIN UNSPLITTABLE FLOW

...And even more network design NP problems...

- Miscellaneous

- MIN BROADCAST TIME
- MIN K-CENTER
- MIN K-CLUSTERING
- MIN K-CLUSTERING SUM
- MIN K-SUPPLIER
- MIN K-MEDIAN
- MIN DIAMETERS DECOMPOSITION
- MAX K-FACILITY DISPERSION
- MIN FACILITY LOCATION
- MAX K-FACILITY LOCATION
- MIN K-SWITCHING NET
- MIN BEND NUMBER
- MIN LENGTH TRIANGULATION
- MIN SEPARATING SUBDIVISION

Minimal Spanning Tree (MST) problem



- Total cost of the links used is a minimum
- All the points are connected together

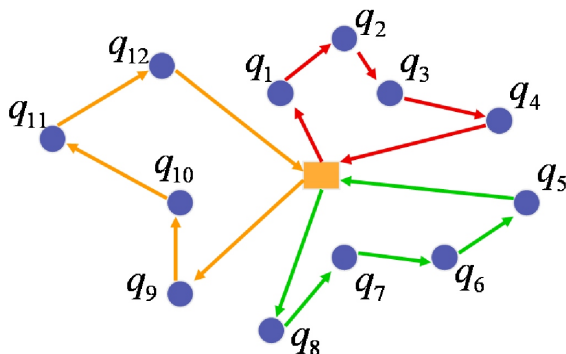
Constraint:

$$x_1 + x_2 + x_6 \leq 1$$

$$x_1 \leq x_3$$

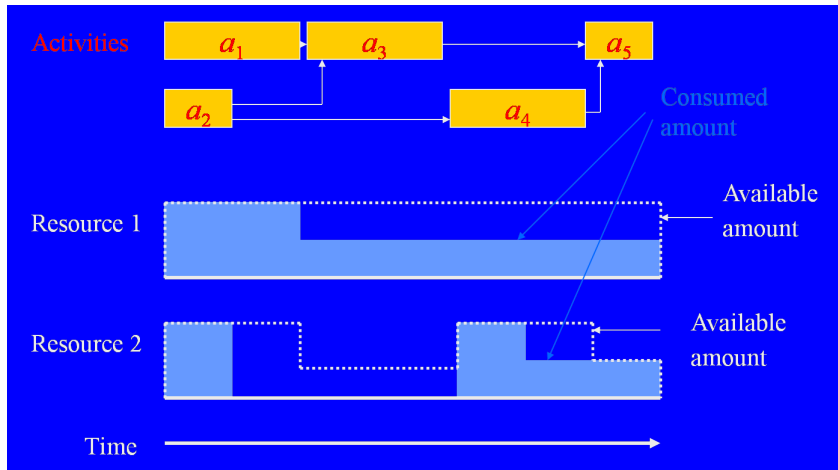
Penalty: 50

Vehicle Routing Problem (VRP) with Time Windows



capacity and **time window** constraints!

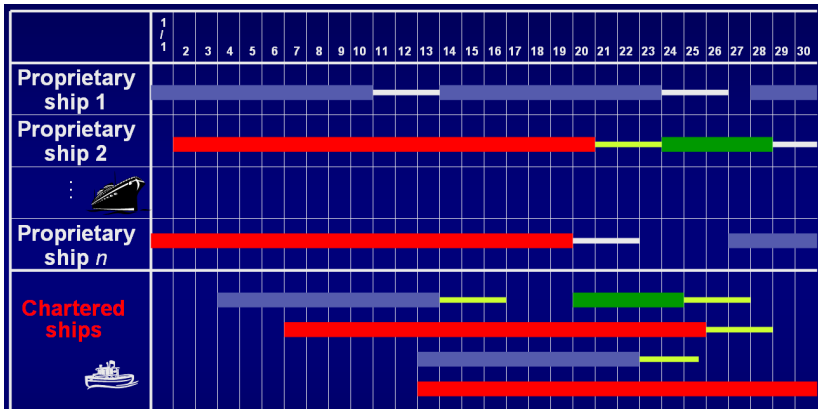
RC PSP – Resource-Constrained Project Scheduling Problem



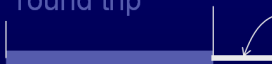
Ship scheduling



Ship Schedule Table



round trip



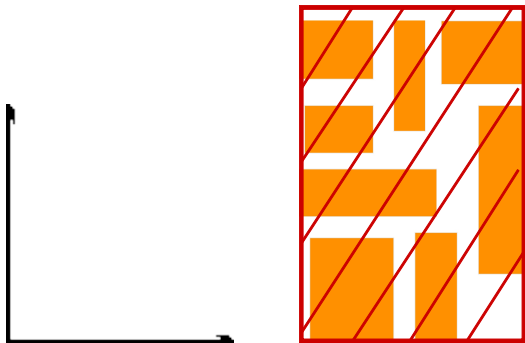
unloading



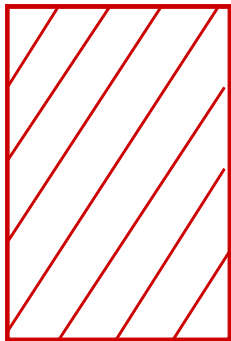
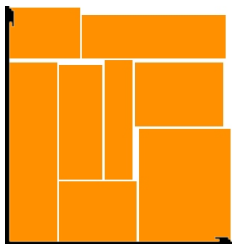
2-Dimensional Packing Problem (2PP)



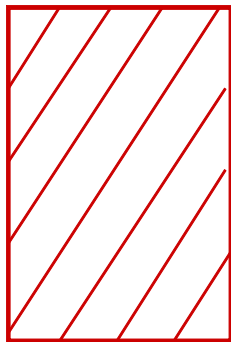
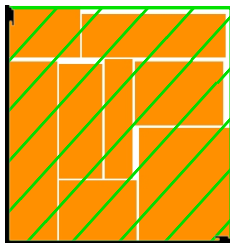
2-Dimensional Packing Problem (2PP)



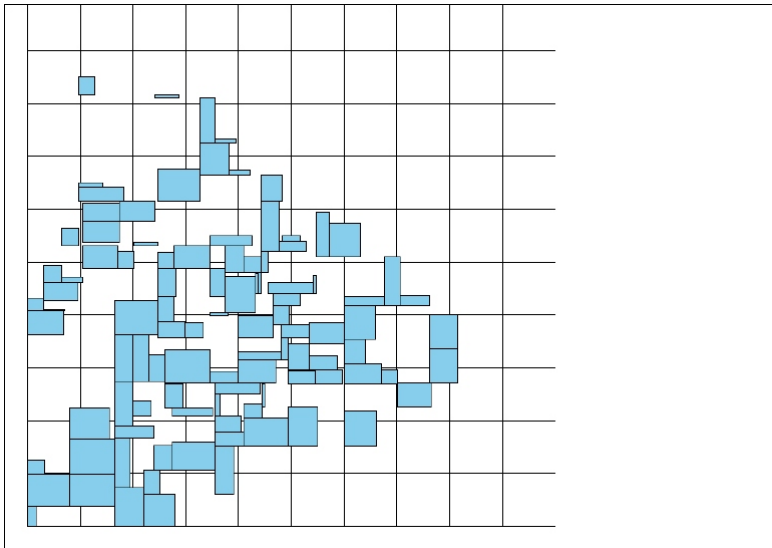
2-Dimensional Packing Problem (2PP)



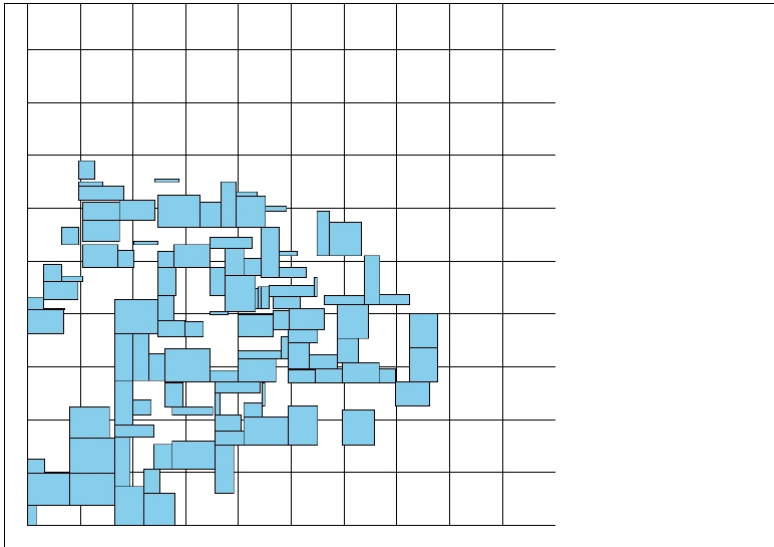
2-Dimensional Packing Problem (2PP)



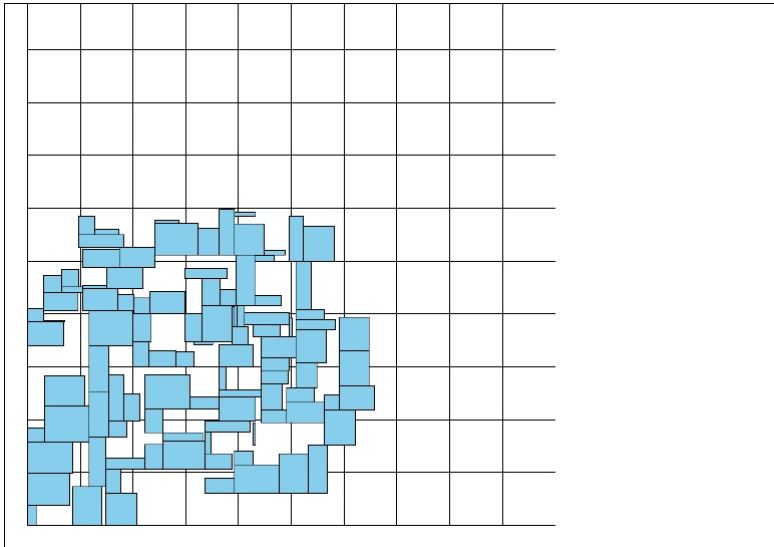
2-Dimensional Packing Problem (2PP)



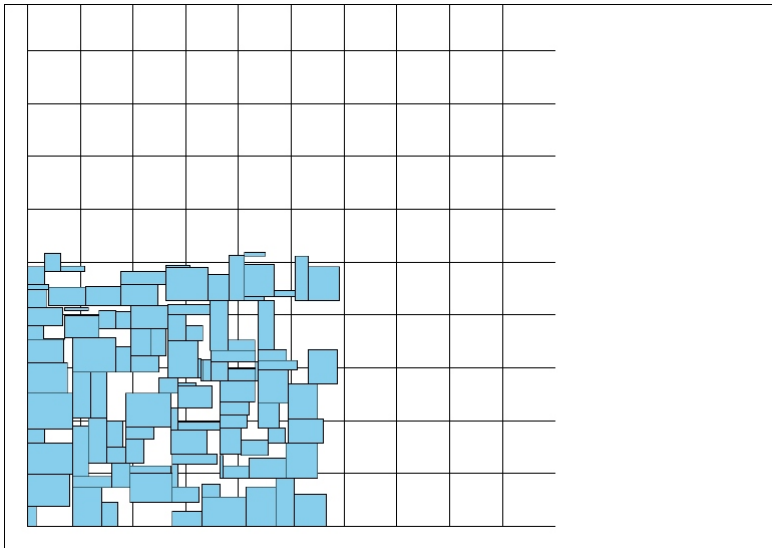
2-Dimensional Packing Problem (2PP)



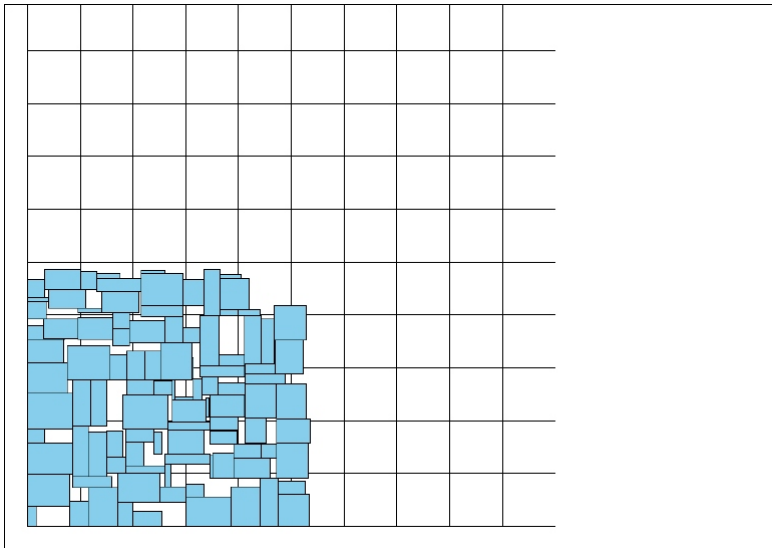
2-Dimensional Packing Problem (2PP)



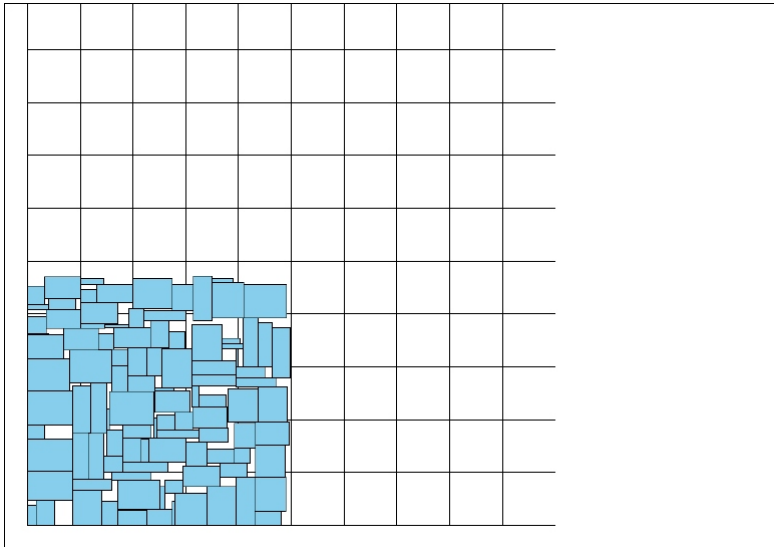
2-Dimensional Packing Problem (2PP)



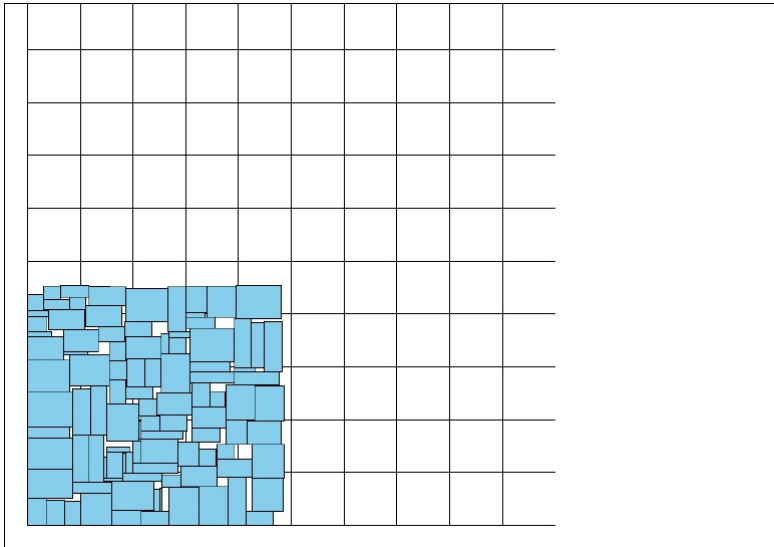
2-Dimensional Packing Problem (2PP)



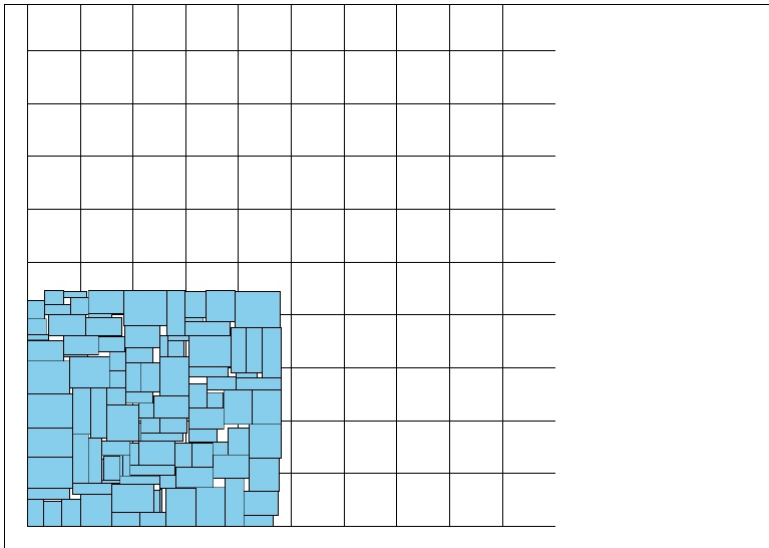
2-Dimensional Packing Problem (2PP)



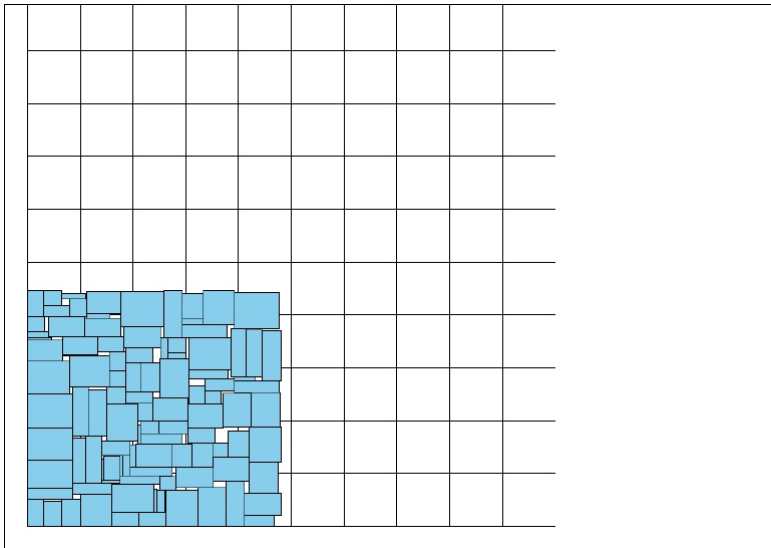
2-Dimensional Packing Problem (2PP)



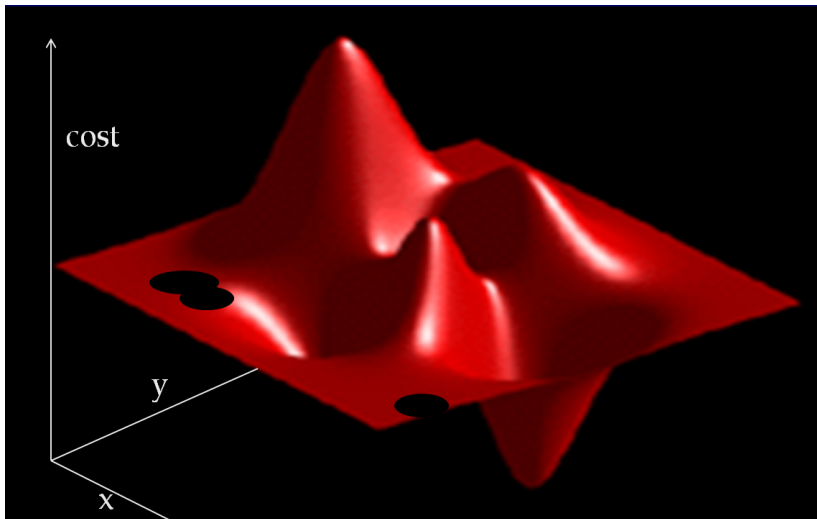
2-Dimensional Packing Problem (2PP)



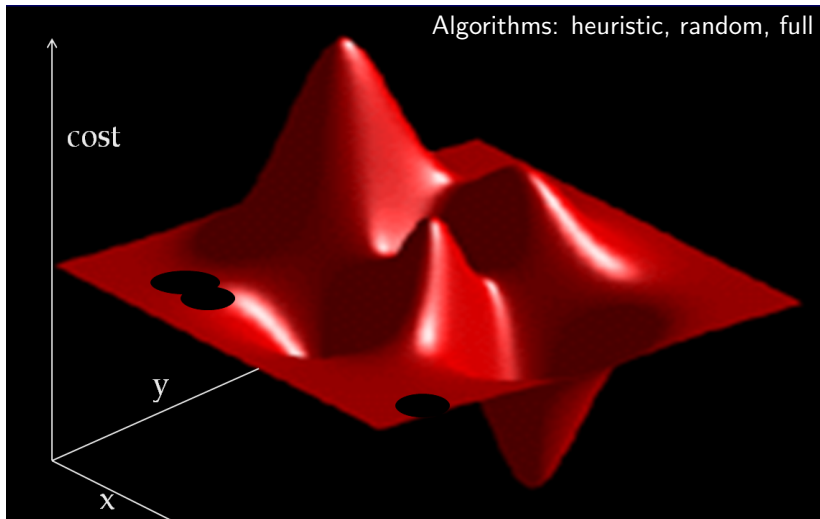
2-Dimensional Packing Problem (2PP)



Optimization landscape

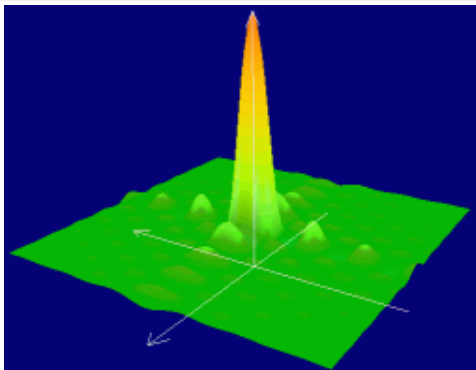


Optimization landscape



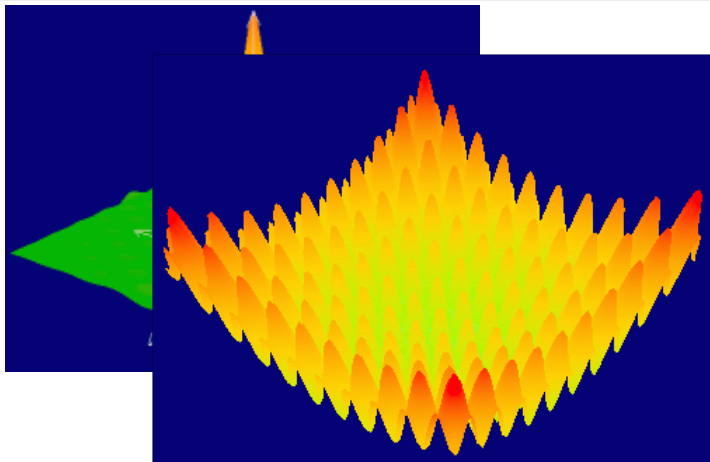
Some landscapes

<http://www.alife.pl/opt/p/index.html>



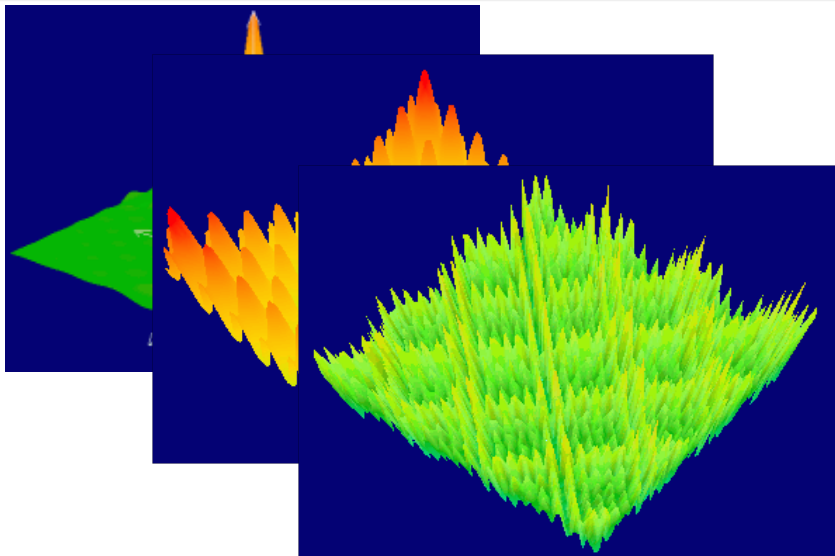
Some landscapes

<http://www.alife.pl/opt/p/index.html>



Some landscapes

<http://www.alife.pl/opt/p/index.html>



Landscapes, GC, mappings, F-D correlation, NFL

