

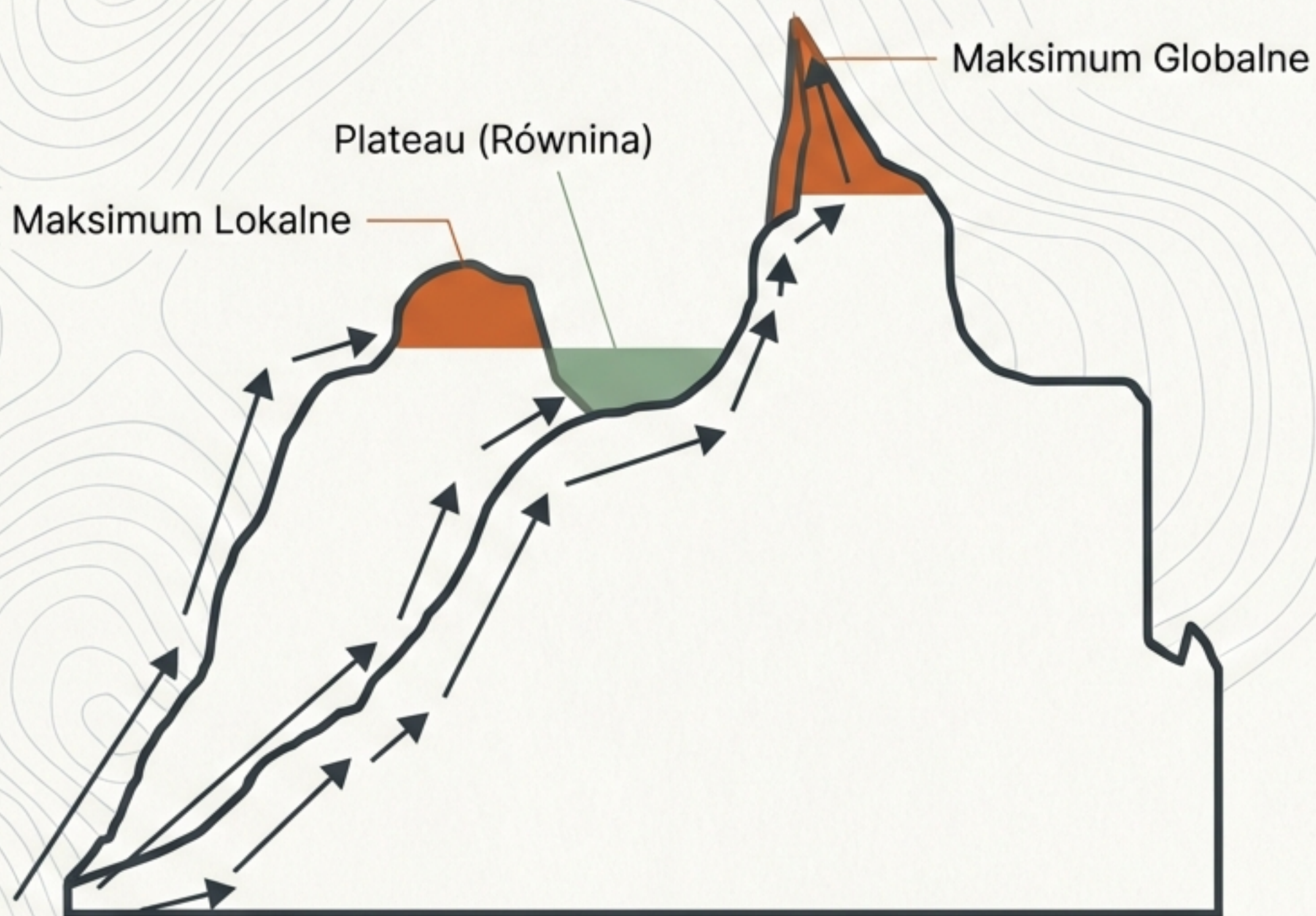
Zaawansowane Algorytmy Heurystyczne

Od wspinaczki po ewolucję –
metody optymalizacji globalnej.

Szukanie heurystyczne II

Autor: Włodzisław Duch, Katedra Informatyki Stosowanej UMK

Krajobraz poszukiwań: szczyty, doliny i mgła



Cel:

Znalezienie punktu, w którym funkcja heurystyczna osiąga wartość ekstremalną (maksimum zysków lub minimum strat).

Wyzwanie:

Problem mgły: Algorytm nie widzi całej mapy. Musi oceniać każdy stan niezależnie, nie znając optymalnej drogi. Lokalne szczyty mogą mylnie sugerować osiągnięcie celu.

Algorytm wspinaczki: Strategia zachłanna



Krok 1: Wybierz najbardziej stromy wznoszący się kierunek

Zasada działania:

1. Oceń stan początkowy.
2. Sprawdź sąsiadów. Jeśli nowy stan jest wyżej (bliżej celu) -> idź tam.
3. Jeśli jest niżej -> zignoruj go.
4. Powtarzaj, aż nie będziesz mógł iść wyżej.

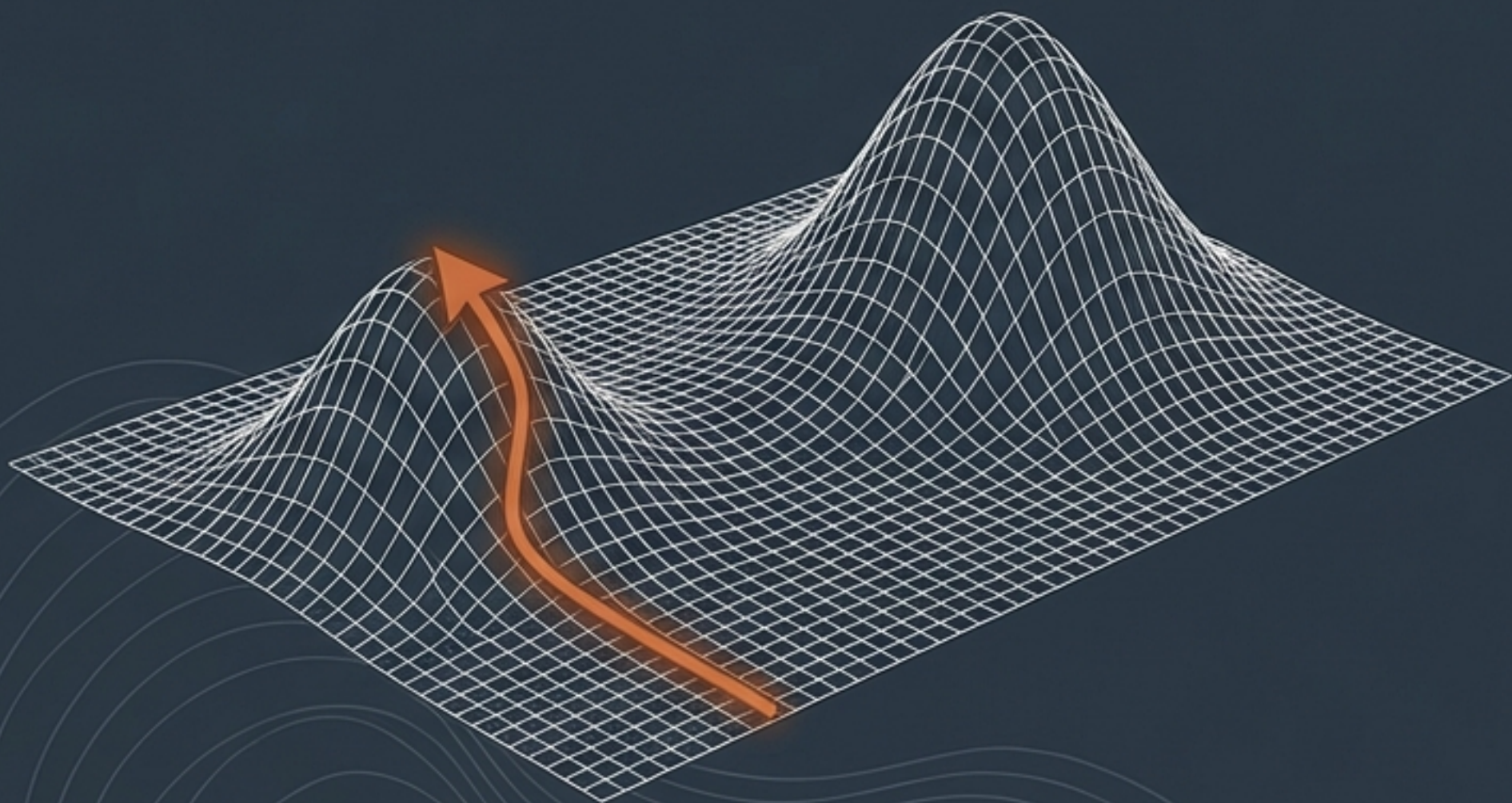
Charakterystyka:

Algorytm pamięta tylko jedną ścieżkę. Jest to odpowiednik metody 'najpierw najlepszy' (best-first) dla parametrów ciągłych.

Wada:

Krótkowzroczność. Jeśli trafisz na impas (identyczne koszty), wybór jest przypadkowy.

Pułapka lokalnego optimum

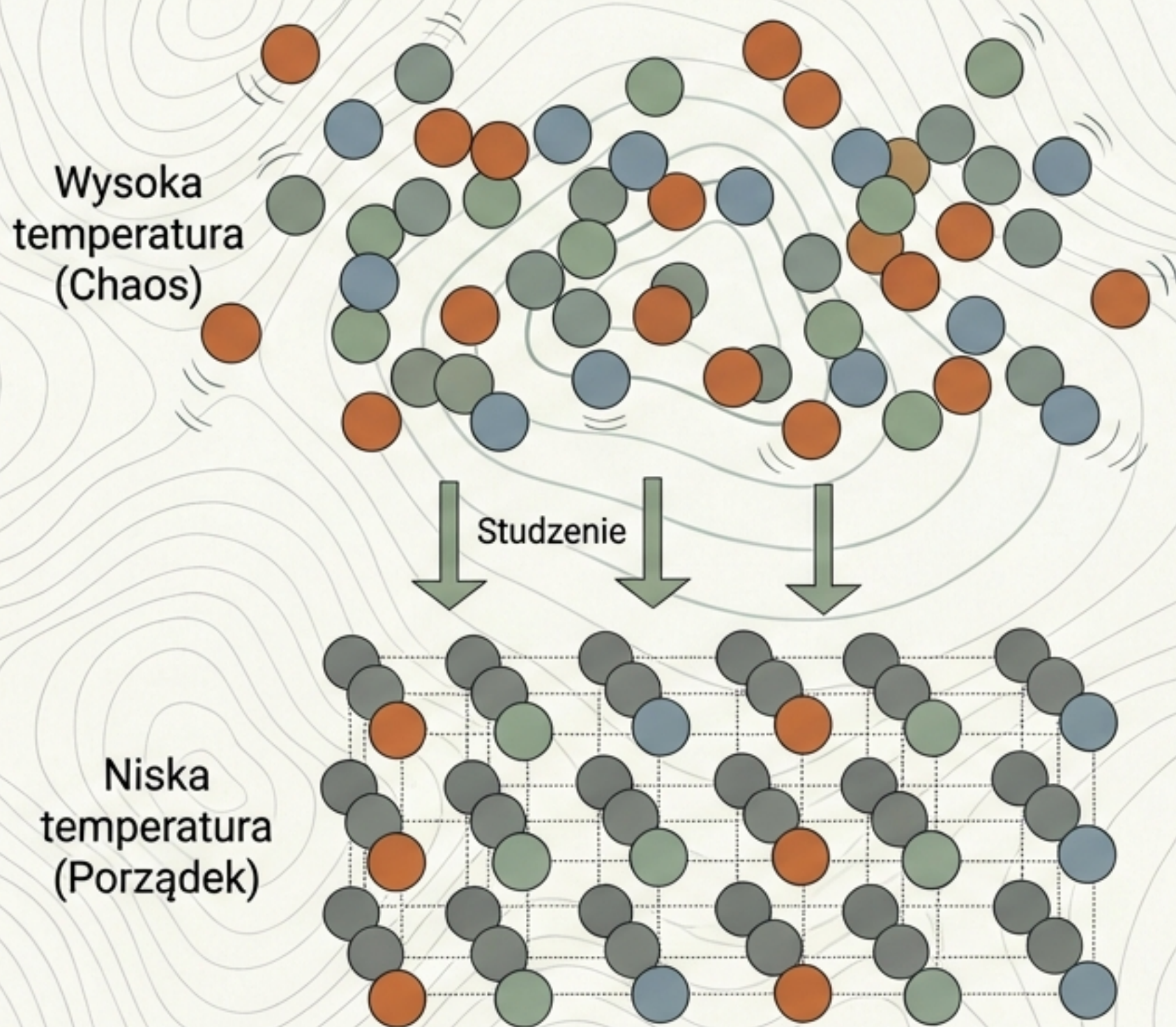


Dlaczego wspinaczka zawodzi?

1. **Lokalne Minima/Maksima:** Zatrzymujemy się na małym pagórku. Każdy krok w bok prowadzi w dół, więc algorytm myśli, że to koniec.
2. **Plateaux (Równiny):** Obszary płaskie, gdzie brak jest kierunku wzrostu – algorytm błądzi bez celu.
3. **Wąskie grzbiety (Ridges):** Trudne do nawigacji prostymi ruchami (północ/południe/wschód/zachód).

Próby ratunku: Wielokrotne starty losowe lub Beam Search (Szukanie wiązką – rozwijanie n najlepszych ścieżek naraz).

Inspiracja z fizyki: Symulowane Wyżarzanie



Analogia metalurgiczna:

Powoli studzone kruształy osiągną konfigurację o minimalnej energii (uporządkowaną). Szybkie chłodzenie prowadzi do defektów.

Kluczowa zmiana:

Aby znaleźć szczyt globalny, czasem trzeba zejść w dół (zaakceptować gorsze rozwiązanie), by wyjść z lokalnej pułapki.

Metoda:

Połączenie losowości (Monte Carlo) ze sterowaniem parametrem temperatury.

Mechanika wyżarzania: Rola temperatury

The Algorithm Logic

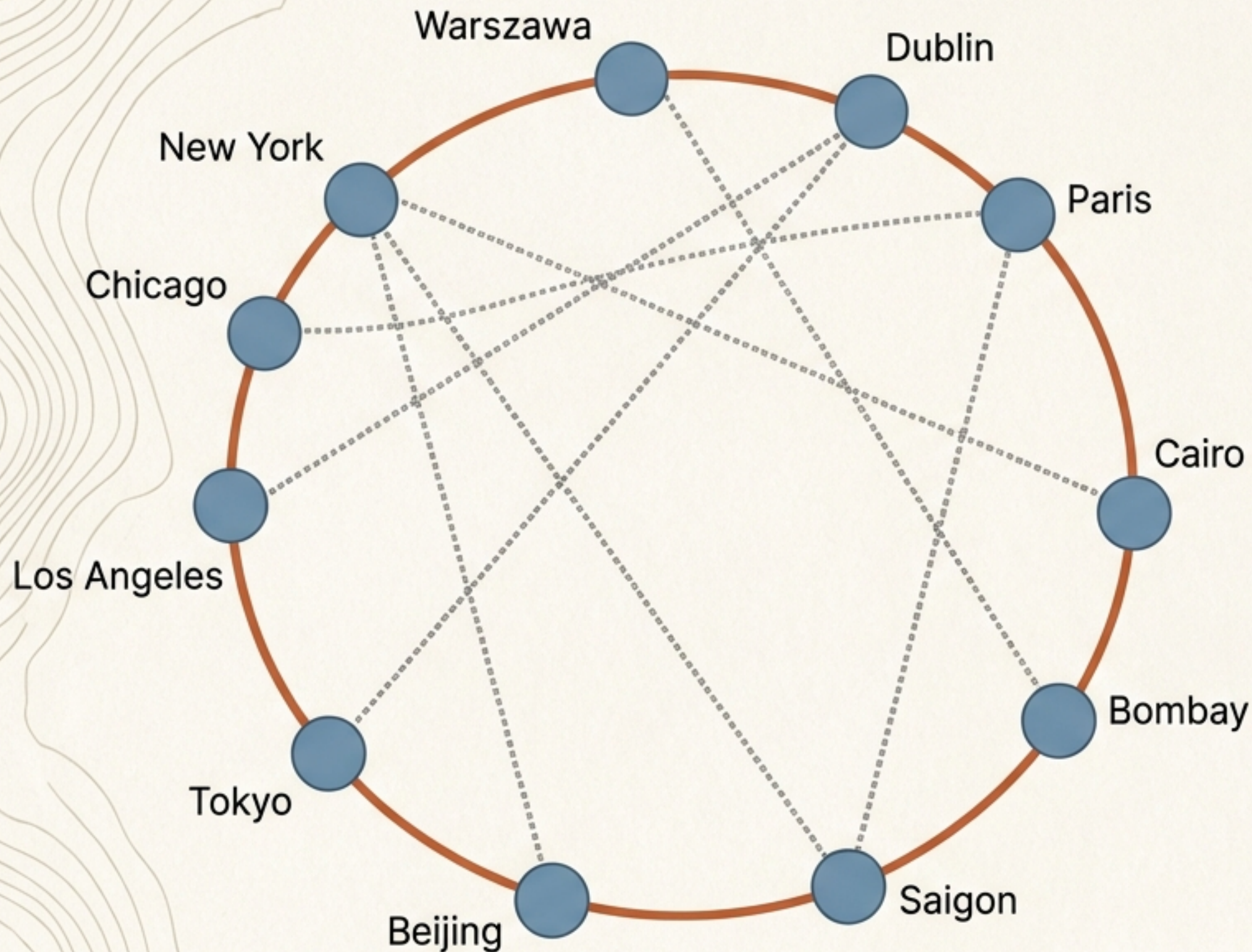
1. **Wysokie T (Eksploracja):** Akceptujemy prawie wszystkie zmiany, nawet te gorsze. Pozwala to na ucieczkę z lokalnych minimów.
2. **Niskie T (Eksploatacja):** W miarę spadku T, akceptujemy tylko poprawę. Algorytm stabilizuje się w optimum.

$$P(\Delta E) \sim \exp(-\Delta E/T)$$

Prawdopodobieństwo zaakceptowania gorszego stanu zależy od różnicy ocen (ΔE) i temperatury (T).

Jest to rozkład Boltzmannna znany z termodynamiki. W przyrodzie układy dążą do minimalnej energii w taki sam sposób.

Zastosowanie: Problem Komijojazera (TSP)



Cel: Odwiedzić 11 miast najkrótszą trasą.

Symulacja:

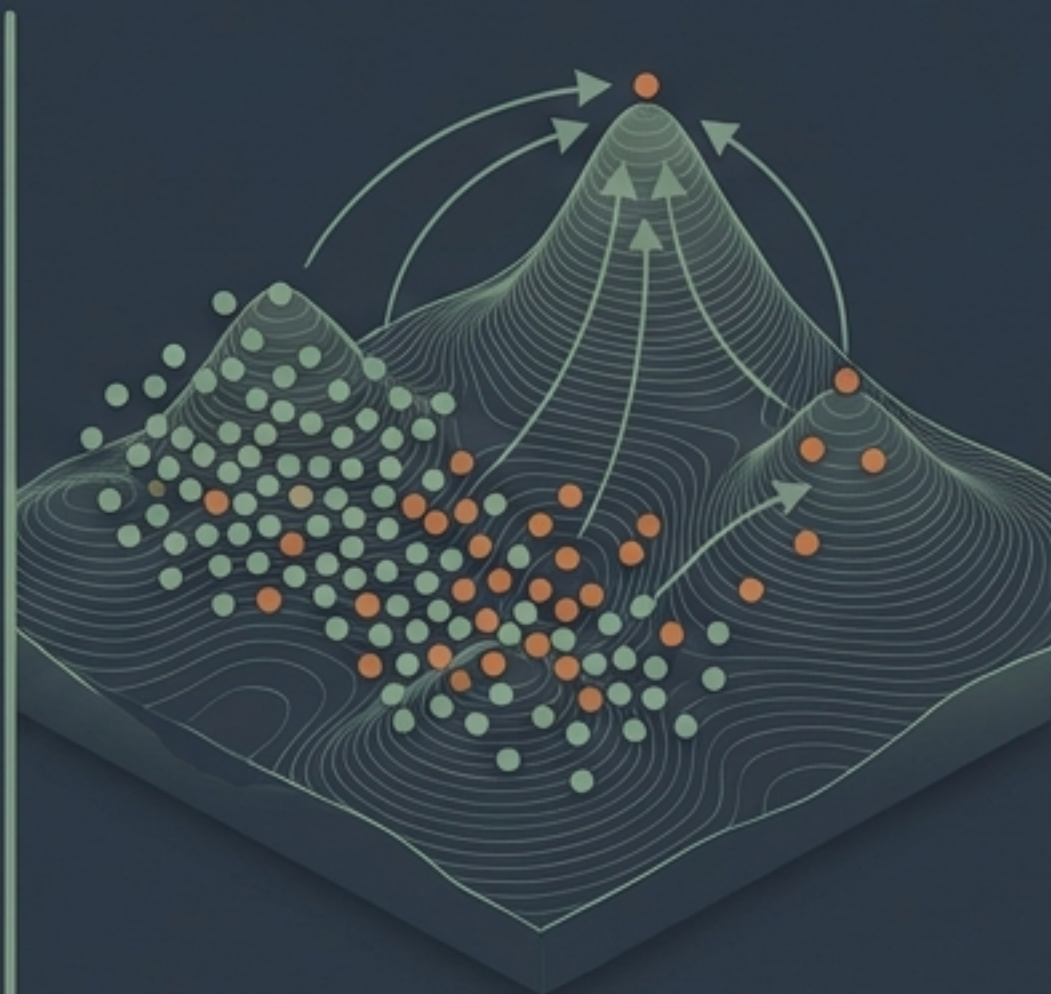
1. Start ($T=100$): Losowa trasa, Koszt = 370.
2. Krok 1 ($T=90$): Zamiana Dublin/Los Angeles. Koszt spada do 340. (Akceptacja: Lepszy wynik).
3. Krok 2 ($T=90$): Zamiana Saigon/Chicago. Koszt rośnie do 377. (Akceptacja: Mimo pogorszenia, bo T jest wysokie).
4. **Koniec** ($T=10$): Koszt = 188. Akceptacja tylko jeśli koszt spada.

Inspiracja z biologii: Algorytmy Genetyczne



Hill Climbing

Zamiast jednego wspinacza, mamy całą populację agentów przeszukujących przestrzeń jednocześnie.



Populacja
(Algorytmy Genetyczne)

Mechanizm Ewolucji:

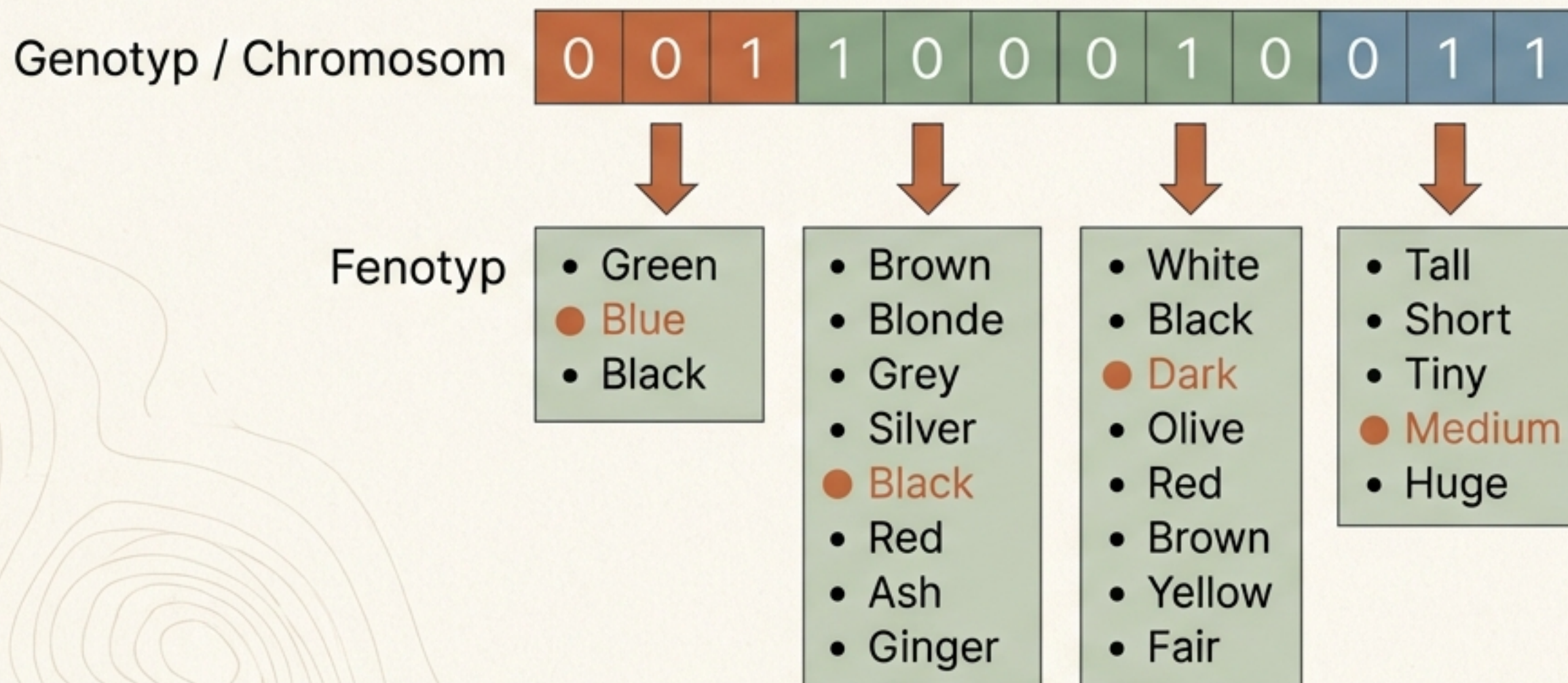
Przetrwanie najlepiej przystosowanych (Survival of the fittest).

Słabe osobniki są usuwane, najlepsze przekazują swoje cechy potomstwu.

"Przyroda nie działa optymalnie, ale często wystarczająco dobrze."

Wniosek: Nie można teoretycznie udowodnić zbieżności do najlepszego wyniku, ale w praktyce metoda jest niezwykle skuteczna w dużych przestrzeniach.

Kodowanie życia: Genotyp i Fenotyp



Chromosom/Genotyp:

Kod genetyczny (np. ciąg binarny). To na nim operuje algorytm.

Fenotyp:

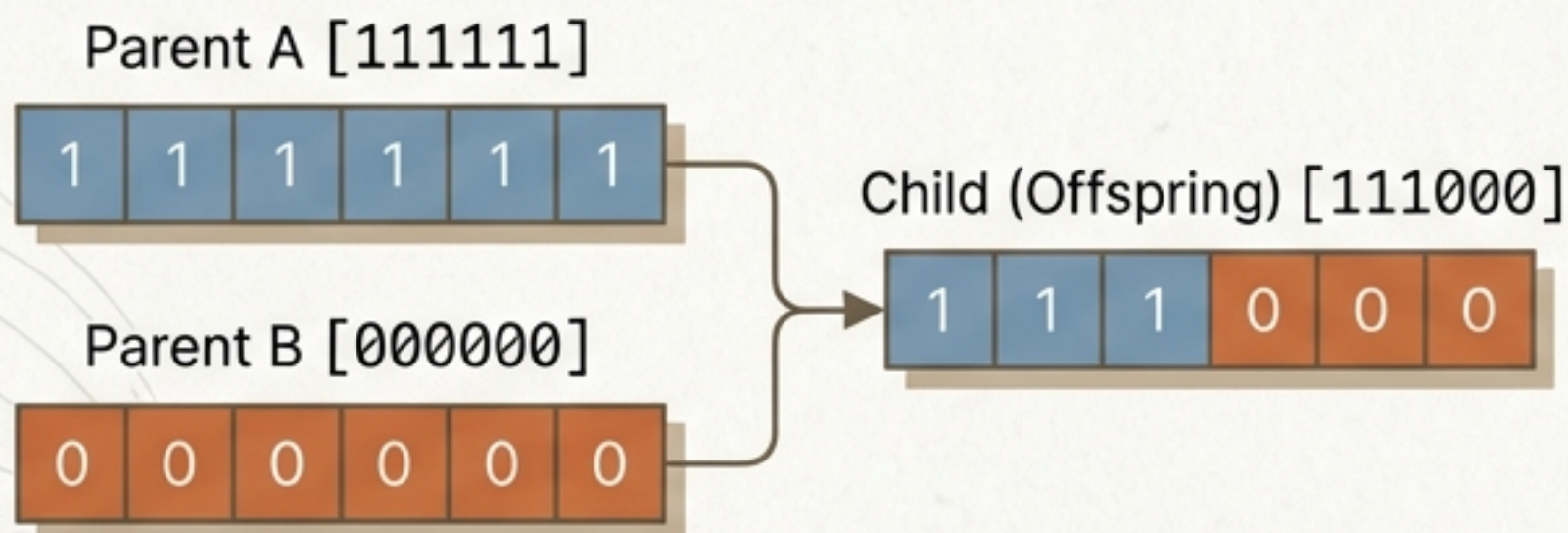
Rzeczywiste cechy wynikające z kodu (np. niebieskie oczy). To jest oceniane przez funkcję przystosowania.

Gen/Allel:

Pojedynczy element kodu i jego warianty.

Operatory ewolucyjne: Krzyżowanie i Mutacja

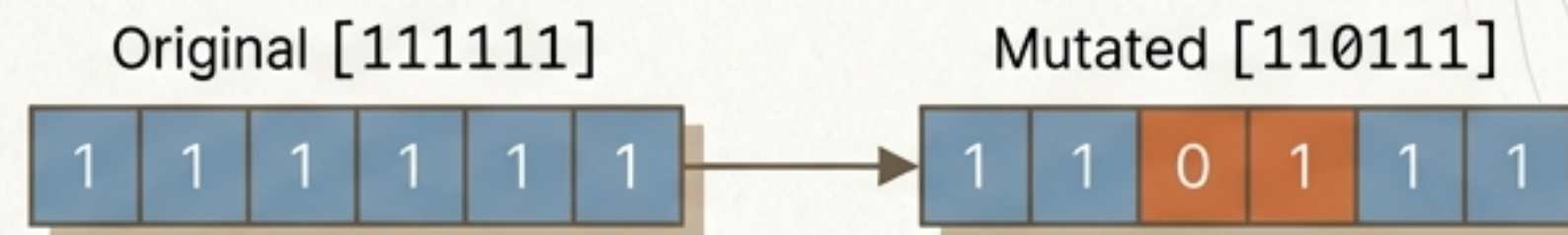
Krzyżowanie (Crossover)



Mieszanie dobrych cech rodziców.

Krzyżowanie: Weź K pierwszych genów od rodzica A i resztę od rodzica B. Tworzy to nowe kombinacje istniejących rozwiązań.

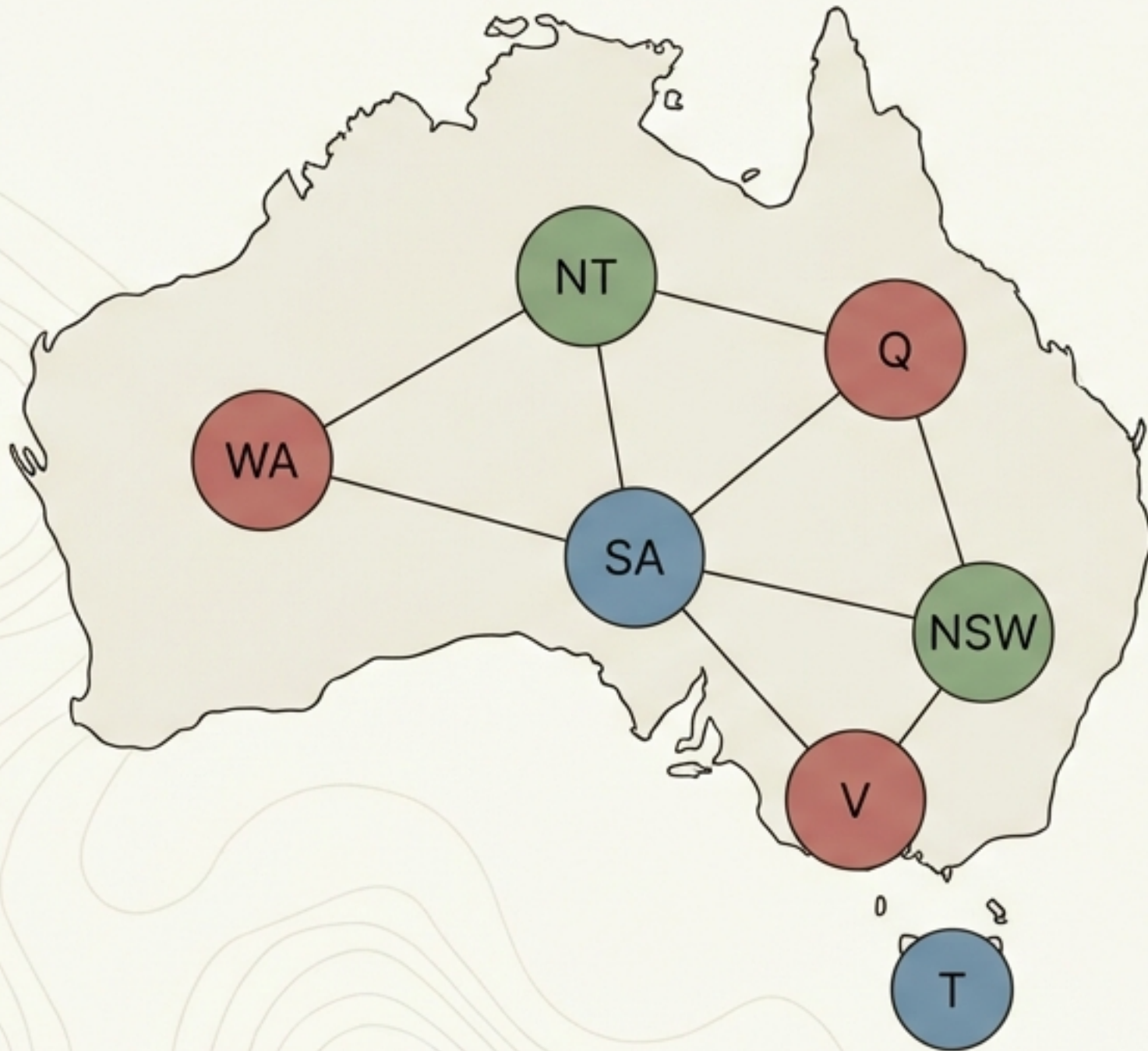
Mutacja (Mutation)



Losowa zmiana (świeża krew).

Mutacja: Losowe odwrócenie bitu. Zapobiega przedwczesnej zbieżności i utknięciu w lokalnym optimum, wprowadzając warianty, których nie mieli rodzice.

Gdy celem jest spełnienie reguł: Problemy CSP



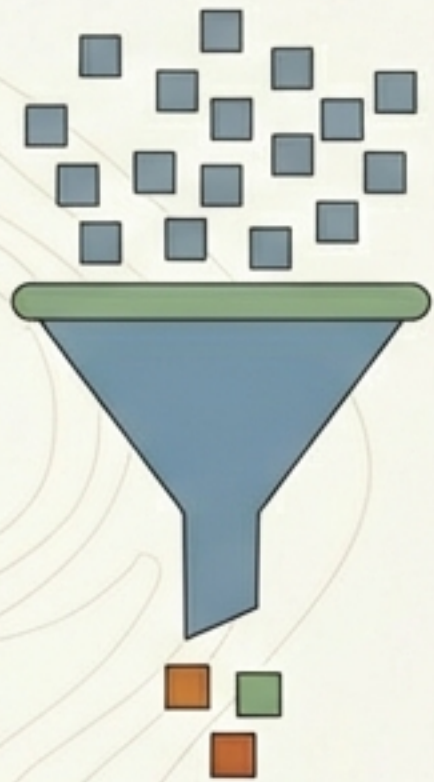
Definicja CSP (Constraint Satisfaction Problem):

Szukanie stanu, który spełnia narzucone ograniczenia, a nie tylko maksymalizuje wynik.

Elementy:

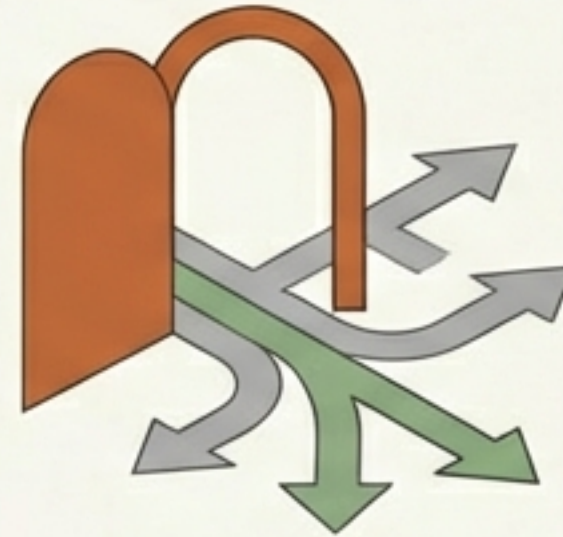
- * **Zmienne (\$X\$):** Regiony na mapie (WA, NT...).
- * **Domeny (\$D\$):** Możliwe wartości (Kolory: Czerwony, Zielony, Niebieski).
- * **Ograniczenia (\$C\$):** "Sąsiednie regiony nie mogą mieć tego samego koloru."

Heurystyki w CSP: Mądry wybór



MRV (Minimum Remaining Values)

Strategia 'Fail-First' (Najpierw najtrudniejsze). Wybierz zmienną, która ma najmniej możliwych ruchów. Jeśli ścieżka jest błędna, lepiej dowiedzieć się o tym od razu, niż po tysiącu kroków.



Least-Constrainting Value

Najmniej ograniczająca wartość. Przy wybieraniu koloru dla danego regionu, wybierz taki, który zamyka najmniej dróg dla sąsiadów. Zostaw sobie otwarte drzwi na przyszłość.

Wynik: Drastyczna redukcja przeszukiwanej przestrzeni poprzez logiczną eliminację.

Siła heurystyki: Porównanie wydajności

Algorytm	Pamięć	Czas	Długość Trasy
Wszerz (BFS)	> 9999	> 3000	> 6000
Najpierw Najlepszy	29	12	3592
A* (A-Star)	15	23	2859

Problem: Szukanie minimalnej drogi pociągiem (połączenia dalekie).

Wniosek: Algorytmy ślepe (BFS) są bezużyteczne przy dużych przestrzeniach – wyczerpują pamięć i czas. Heurystyka A* znajduje optymalną trasę przy minimalnym zużyciu zasobów.

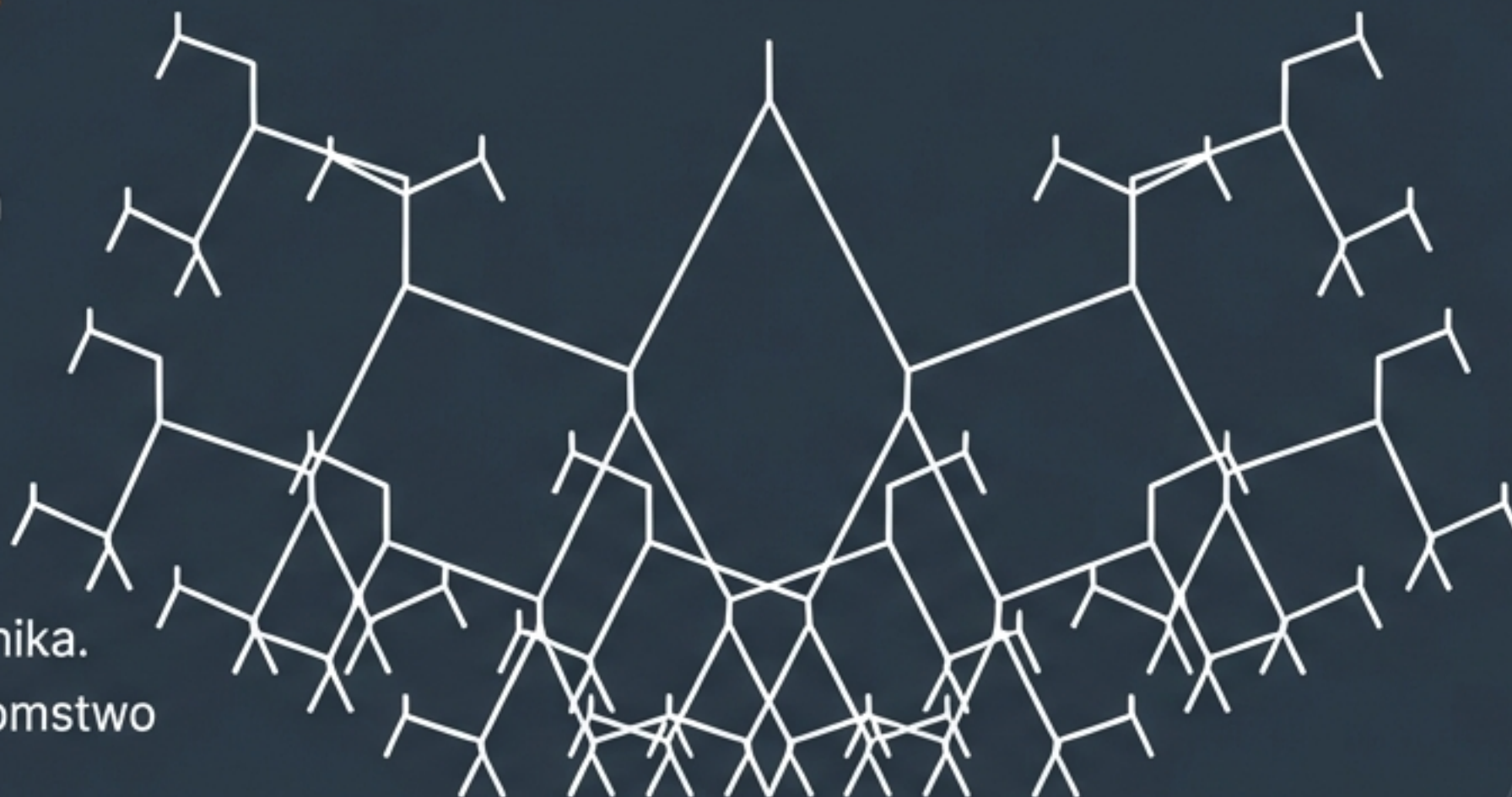
Biomorfy i Sztuka Genetyczna

Eksperyment Dawkinsa:

Struktury wyewoluowane przez selekcję kierowaną przez człowieka (estetykę), a nie funkcję przetrwania.

Proces:

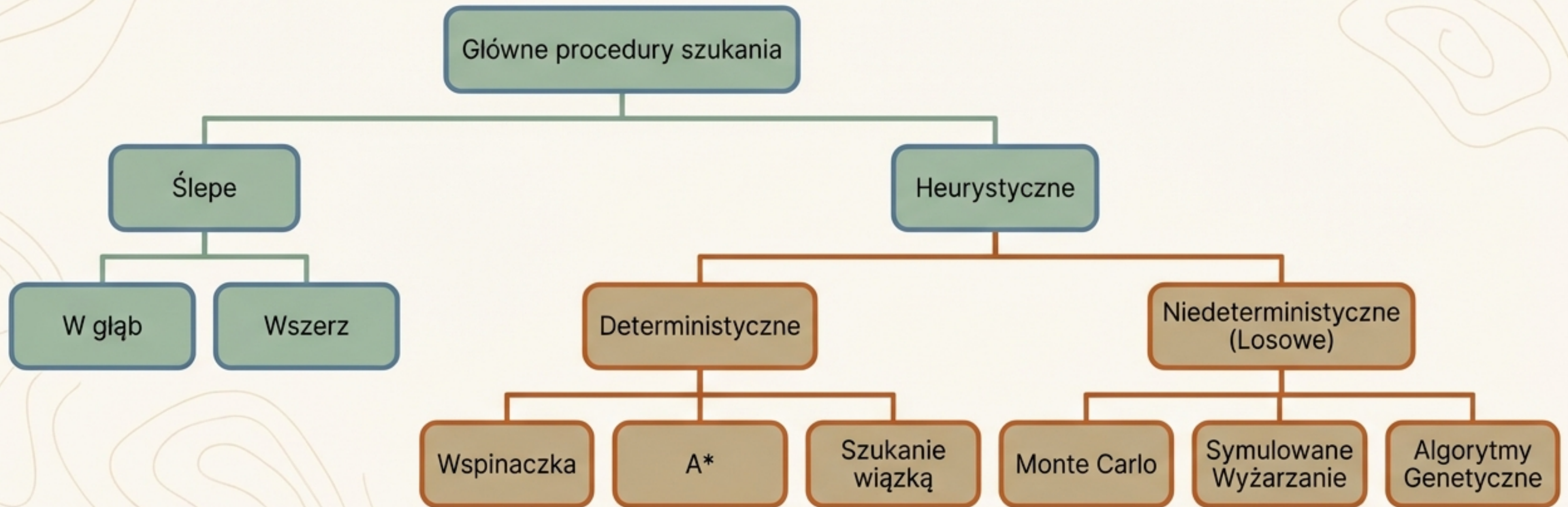
1. Człowiek wybiera "najciekawszego" osobnika.
2. Komputer generuje potomstwo z losowymi mutacjami (długość gałęzi, kąt).
3. Powstają skomplikowane, organiczne kształty przypominające owady lub rośliny.



Zastosowanie:

Projektowanie robotów (nauka chodzenia), sztuka generatywna.

Mapa metod poszukiwania



Podsumowanie: Nie ma jednego idealnego algorytmu ('No Free Lunch'), ale zaawansowane meta-heurystyki pozwalają rozwiązywać problemy, które są matematycznie nierozwiązywalne dla metod siłowych.