

## Etap 4 - logika symulacji

### Diagramy sekwencji

- StateCopulate

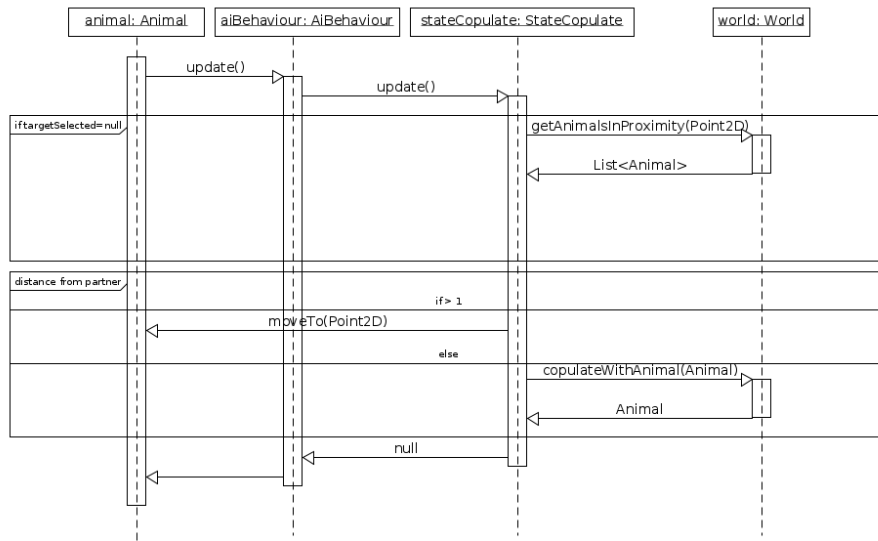


Figure 1: copulate

- StateHunt
- StateLookForFood

Jak można zauważyć stany `Copulate` i `Hunt` mają bardzo podobną strukturę - obiekt stanu wysyła do `World` zapytanie o listę innych zwierząt w pobliżu, którą filtrują wg. swoich predykatów. Tak podobną funkcjonalność można by wydzielić do oddzielnej klasy.

Zmiana aktualnego stanu bez wypychania nowego stanu na stos będzie realizowana przez zwracanie nowego stanu przez `IAiState::update()`.

### Diagramy aktywności

- Hunt
- Copulate

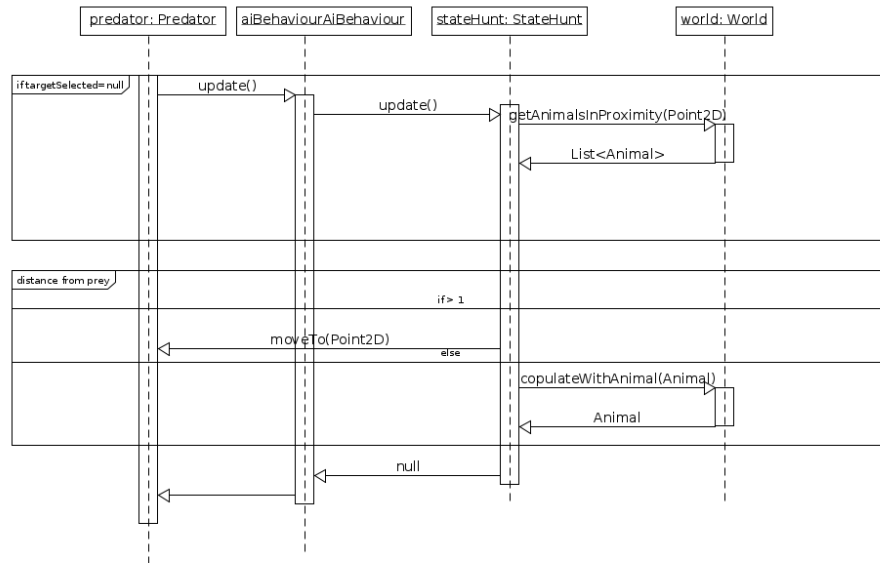


Figure 2: hunt

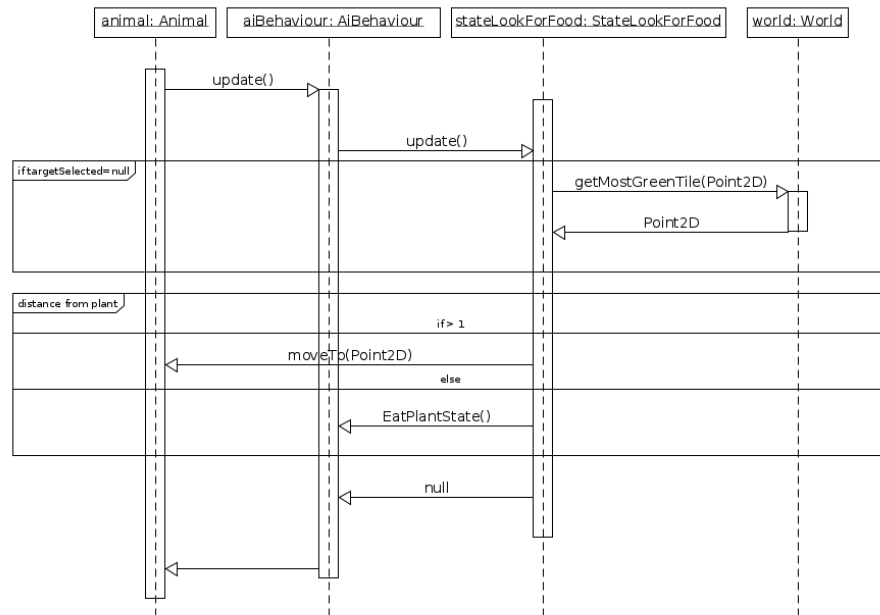


Figure 3: lookforfood

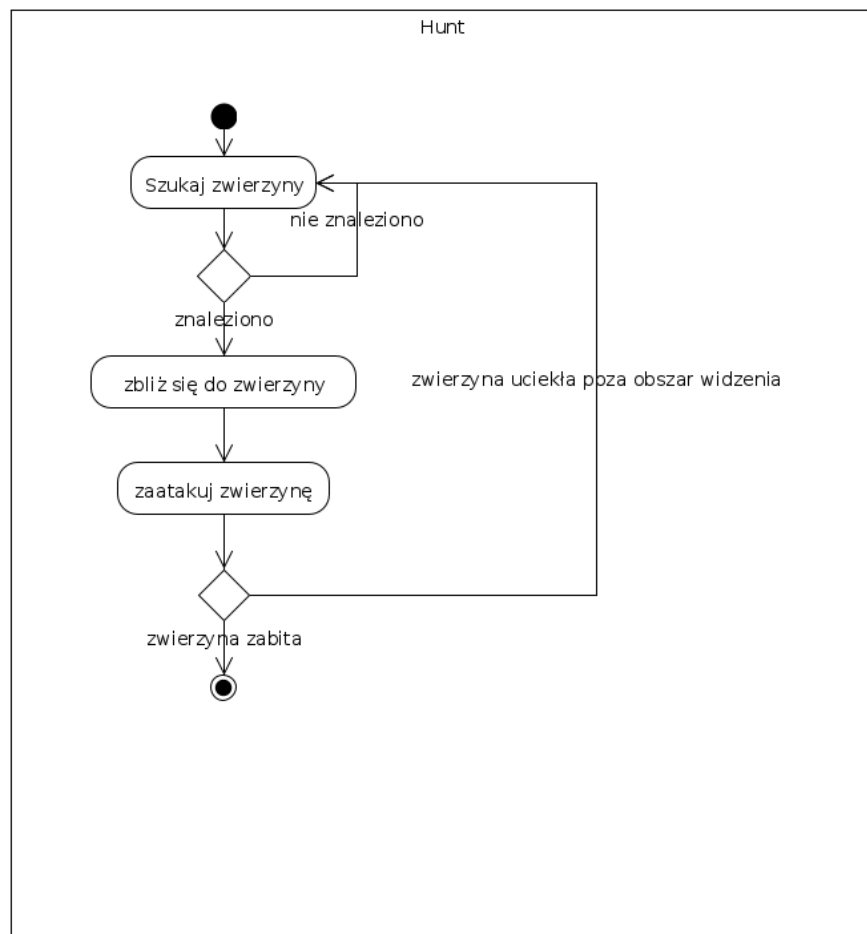


Figure 4: hunt

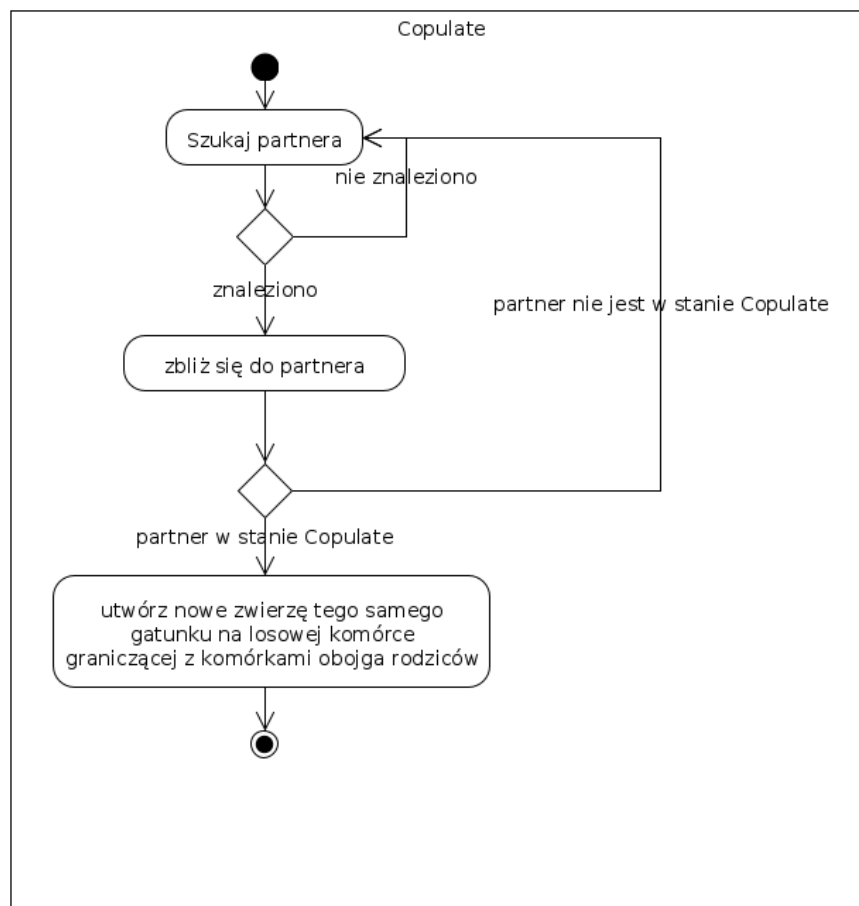


Figure 5: copulate

## Diagramy maszyny stanów

Zachowanie zwierząt będzie używało podejścia automatu ze stosem. Przykład: Podczas normalnego funkcjonowania (zwykła maszyna stanów) pewne zwierzęta będą celem polowań innych zwierząt. W takim wypadku, gdy zwierzę zauważy lub usłyszy drapieżnika znajdującego się w odległości mniejszej niż pewna określona odległość graniczna, zwierzę powinno wejść w stan ucieczki, a po jego zakończeniu (oddaleniu się od drapieżnika na odległość większą niż graniczna) powinno wrócić do stanu poprzedniego. Zostanie do tego wykorzystany stos; wejście w stan ucieczki będzie wepchnięciem go na stos, a wyjście z niego będzie wypchnięciem go ze stosu.

- Antelope

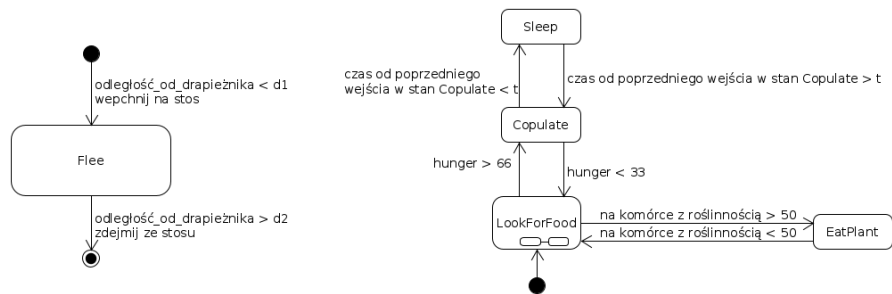


Figure 6: antelope

- Hippo

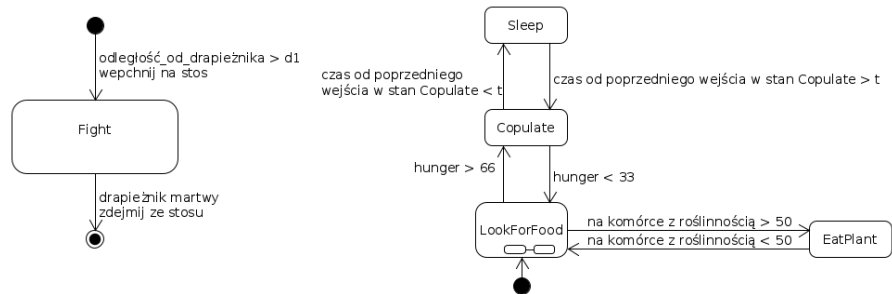


Figure 7: hippo

- Wolf
- Lion
- Crocodile

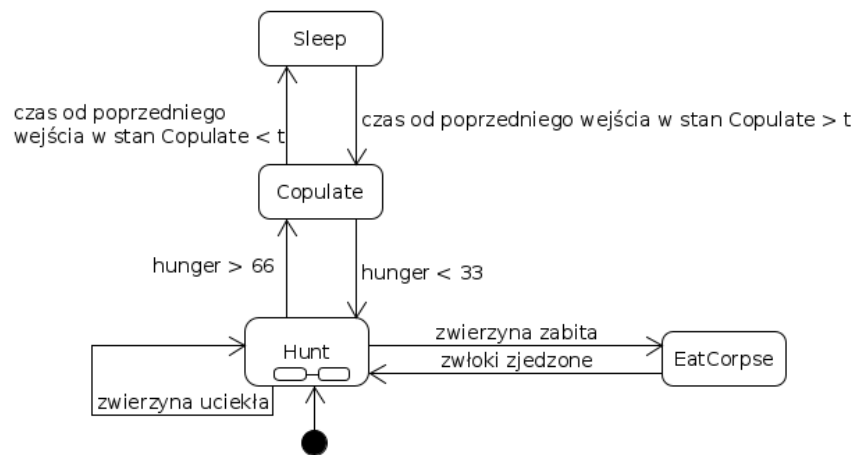


Figure 8: wolf

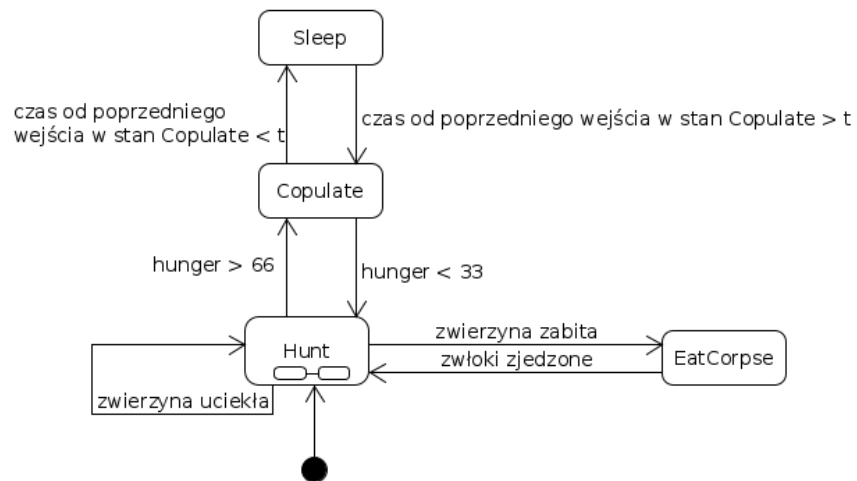


Figure 9: lion

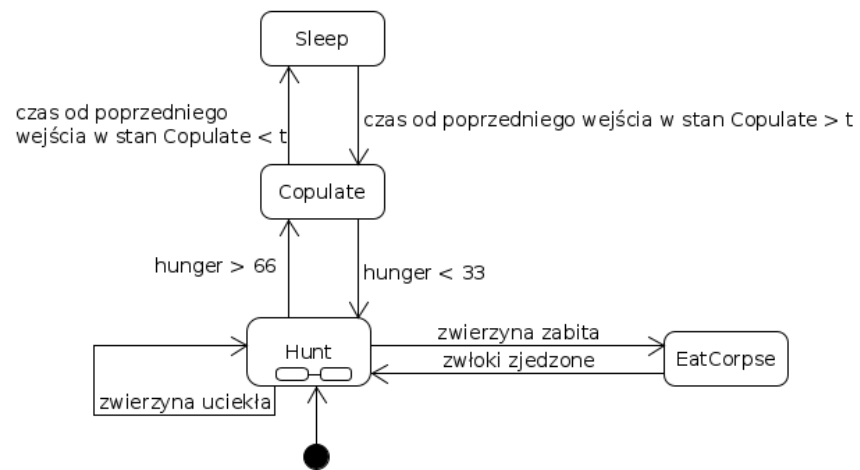


Figure 10: crocodile