Dokumentacja projektu z przedmiotu Podstawy Sztucznej Inteligencji

Magiczne kwadraty

Jan Wiśniewski, Adam Mościcki, Edwin Jarosiński

21 stycznia 2015

1 Cel projektu

Celem projektu jest stworzenie aplikacji pozwalającej na generowanie „magicznych kwadratów" przy użyciu algorytmów ewolucyjnych.

2 Założenia wstępne

Przyjmujemy następującą definicję magicznego kwadratu: tablica składająca się z n wierszy i n kolumn ($n > 2$), w którą wpisano n^2 różnych nie powtarzających się dodatnich liczb naturalnych (od 1 do n^2) w ten sposób (użytkownik ma możliwość wyboru definicji), że:

- suma liczb w każdym wierszu, w każdej kolumnie i w każdej przekątnej jest taka sama,
- suma liczb w każdym wierszu, w każdej kolumnie jest taka sama,

i wynosi $\frac{n^2 \cdot (n^2+1)}{2 * n}$

3 Funkcja dopasowania

Metoda prezentuje się następująco:

- wyliczenie oczekiwanej wartości sum w kolumnach, wierszach i przekątnych: $s = (1 + n^2)\frac{n}{2}$
- policzenie sum w kolumnach, wierszach i przekątnych(jeżeli przekątne są ważne dla typu kwadratu)
- policzenie dla każdej z tych sum wartości absolutnej z różnicy tej sumy i wartości oczekiwanej
- zsumowanie wartości absolutnych

4 Algorytm ewolucyjny

Zastosowany przez nas algorytm to $\mu + \lambda$. Jego pseudokod jest następujący:

1. Wylosuj populacje początkową P o rozumiarze μ osobników postaci $< x, \sigma >$, gdzie x oznacza osobnika, a σ prawdopodobienstwo jego mutacji (wylosowane z przedziału podanego przez użytkownika),
2. Wybierz z P za pomocą odpowiedniej strategi λ osobników, którzy będą się rozmnażać jako populacje tymczasową T.
3. Zastosuj krzyżowanie dla populacji T, oraz mutację osobnika z prawdopodbieństwem σ dla kwadratu (każdego z osobna) - wynik pouplacja R.

4. $P := P \cup R$.

5. Jeśli znaleziono magiczny kwadrat zakończ algorytm.

5 Operator krzyżowania

Najważniejszą decyzją do podjęcia przy realizacji algorytmu ewolucyjnego jest wybór operatora krzyżowania. Jako że w podstawowej reprezentacji magiczny kwadrat jest macierzą, a wartość funkcji dopasowania zależy od sum kolumn wierszy i przekątnych, należy bardzo dokładnie przeanalizować dostępne opcje. Po rozważeniu różnych możliwości zdecydowaliśmy się na implementację dwóch różnych operatorów:

5.1 Zamiana przekątnych

1. utworzenie dziecka - przepisanie pól rodzica A
2. wybór jeszcze nie wybranego pola rodzica B znajdującego się na jednej z jego przekątnych
3. przepisanie wybranego pola z rodzica B do odpowiadającego mu pola dziecka
4. jeżeli w dziecku występuje druga taka sama liczba jak ta właśnie pobrana od rodzica B to jest ona podmieniana na liczbę która została przez nią nadpisana
5. jeżeli rodzic B posiada jakieś nie wybrane pola na przekątnych to skok do punktu nr 2
6. koniec

5.2 Zamiana wierszy albo kolumn

Opisany algorytm operuje na kolumnach. Zamiana wierszy jest dokonywana analogicznie.

1. utworzenie dziecka - przepisanie pól rodzica A
2. losowy wybór $\lfloor \frac{N}{2} \rfloor$ kolumn
3. wybór jeszcze nie wybranego pola rodzica B znajdującego się w wybranych kolumnach
4. przepisanie wybranego pola z rodzica B do odpowiadającego mu pola dziecka
5. jeżeli w dziecku występuje druga taka sama liczba jak ta właśnie pobrana od rodzica B to jest ona podmieniana na liczbę która została przez nią nadpisana
6. jeżeli rodzic B posiada jakieś nie wybrane pola na wybranych kolumnach to skok do punktu nr 3
7. koniec

Niezależnie od metody krzyżowanie rodziców wartość σ dla dziecka jest wyliczana jako średnia arytmetyczna z wartości σ obojga rodziców.

6 Operator mutacji

Dostępne są 2 operatory testowania:

1. zamiana dwóch punktów,
2. zamiana dwóch kolumn.

Warto zauważyć że 2 z operatorów nie zamienia sumy w kolumnach i wierszach (a tylko na przekątnych), zmiana wartości funkcji przystowania w tym przypadku jest więc zależna tylko od operatora krzyżowania.

7 Wybór osobników do krzyżownia

Selekcja nastepuje w trzech możliwych sposobach:

- Rankingowa: Osobniki są wybierane z prawdopodobieństwm proporcjonalnym do rankingu ich funkcji przystowania.
- Ruletki: Ta metoda jest zmodyfikowaną wersją standardowej wersji, polega na wyborze osobników z pradopodbieństwem $1 - \frac{przystowanie(osobnik) - przystosowanie(najlepszy)}{przystosowanie(osobik)}$.
- Najlpeszych: Brane jest μ najlepszych osobików (jeżeli μ jest wieksze od wielkości populacji, to branie jest rozpoczynane od końca).

Praktyka pokazuje, że ostatni z tych typów daje najlepsze rezultaty, ponieważ często jest tak, że jedna kluczowa zamiana potrafi osobnika bardzo złego zamienić na bardzo dobrego.

8 Optymalizacje

Użytkownik ma do wyboru 2 optymalizacje. Pierwsza polega na dodaniu nowych osobników co którąś iterację (czasem pozwala to wyjść z lokalnego maksimum). Drugie jest przyspieszeniem symulacji poprzez ogranicznenie populacji do jakichś rozmiarów (dzięki czemu oszczędzamy czas nie przetwarząjac słabo przystosowanych osobników).

9 Testowanie

Testowaliśmy naszą symulację z parametrami początkowymi. Najlepsze wyniki uzyskiwaliśmy przy wykorzystaniu metody Najlepszych jako użytej metody selekci, zamiany losowych punktów jako metody mutacji oraz wymiany przekatnych jako metody reprodukcji. Skuteczność działania naszego algorytmu często poprawia dodanie nowych osobników do populacji co jakiś czas.
Sto prób dla kwadratu z liczącymi się przekątnymi o rozmiarze 3, z parametrach μ równemu 500 oraz λ równemu 1000, σ stałą równa 80% dla selekcji najlepszych przy mutacji zamieniającej 2 pola zakończyło się sukcesem w 100 przypadkach (3 z nich wychodziły poza 10 iteracje).
Jeden z otrzymanych kwadratów:
4 8 3
2 6 7
9 1 5
20 prób dla kwadratu o rozmiarze 4 przy tych samych parametrach dało sukces w 5 przypadkach.
Jeden z otrzymanych kwadratów:
7 6 12 9
14 15 1 4
2 3 16 13
11 10 5 8

10 Wnioski

Dla tablic o wielkości nie przekraczającej 5 da się bez większych problemów generować magiczne kwadraty (czasem po kilkunastu próbach). Z powodu problemów ze znalezieniem optymalnego operatora krzyżowania stosowanie algorytmów ewolucyjnych do tworzenia magicznych jest mało optymalne i nie powinno stanowić większego problemu znalezienie strategii heurystycznej radzącej sobie z tym problemem zdecydowanie lepiej.