

РЕАЛИЗИРАНЕ НА ЯДРО ЗА ГЕНЕТИЧНА ОПТИМИЗАЦИЯ В РЕАЛНО ВРЕМЕ ЧРЕЗ ЦИФРОВ СИГНАЛЕН ПРОЦЕСОР

Доц. д-р Борко Ганев Боянов

Гл. ас. Емил Тодоров Тодоров

Технически Университет - Варна, България

Results of development of a real-time software kernel intended for genetic algorithm optimization/adaptation are presented in this paper. Genetic algorithm (GA) is relatively new method for optimization or adaptation, which provides stochastic search over the parameter space, guided by fitness evaluation towards specific goal. Although relatively slow, GA can handle complex optimization problems, especially when the goal is to find near-best extreme in multimodal function domain. GA features also global search from many parameter space points, capability to escape from local extreme, and absence of derivatives. Whereas most publications present results from computer simulations, this paper discusses real-time GA optimization / adaptation systems based on digital signal processor (DSP). Specifics of DSP implementation of basic genetic operations such as genetic crossover, mutation, parent selection, and replacement are discussed. Computational efficiency of the developed software kernel for ADSP2181 digital signal processor is evaluated.

I. ВЪВЕДЕНИЕ

Генетичните алгоритми се използват в различни етапи при разработването и функционирането в реално време на различни радиотехнически системи. На етапа на проектирането генетичните алгоритми могат да се използват за оптимизиране на параметрите на дадената система, докато при функционирането в реално време те могат да се използват за адаптиране на системата към външни въздействия [1]. Един типичен пример за адаптация е управлението на диаграмата на насоченост на многоелементна антенна решетка чрез управление на амплитудите и фазите на сигналите на отделните елементи. Такава задача е решена в [2] по отношение на управление на нулата на диаграмата на насоченост на многоелементна антенна решетка, като алгоритъмът за генетично адаптиране се реализира с персонален компютър (ПК). Предимствата на използването на ПК са по-лесното разработване на софтуера за адаптиране и документирането на резултатите. В същото време обаче може да се посочи, че използването на ПК е само междинен етап и за реализиране на практически приложима система той не е подходящ от гледна точка на стойност и размери.

В настоящата работа се представят резултатите от разработването на специализиран хардуер и софтуер, подходящи за реализиране на генетичен алгоритъм за адаптация (ГАА) в реално време, базиран на процесор за цифрова обработка на сигнали. Първоначалното предназначение е за адаптация на многоелементни антенни решетки, но поради достигнатата универсалност хардуерът и софтуерът могат да се разглеждат като ядро за генетично оптимизиране.

II. ОБЩИ СВЕДЕНИЯ ЗА ОПТИМИЗАЦИЯ С ИЗПОЛЗУВАНЕ НА ГЕНЕТИЧЕН АЛГОРИТЪМ

Генетичните алгоритми за оптимизация / адаптация симулират процесите на развитие, които се наблюдават в природата. Генетичният алгоритъм се състои от етапи: оценка на пригодността, генетична селекция, генетични операции и формиране на нова популация, които се повтарят циклично. Популацията се състои от определен брой индивиди, всеки представен чрез своя хромозом, кодиращ всички черти на индивида. Приложено към физическия проблем за адаптация на многоелементна система в реално време кодирането се състои в представянето съвкупността на всички регулируеми параметри в двоичен вид и записването им в един стринг.

2.1. Оценка на пригодността.

Оценката на пригодността на всеки индивид от популацията се извършва като се използва съответния хромозом, от който чрез обратно преобразуване и предаване към физическата система се получава конкретната реализация на системата. Изходният сигнал на физическата система се използва като оценка на пригодността на конкретната реализация.

2.2. Генетична селекция.

Една от основните идеи на оптимизацията чрез генетичен алгоритъм е, че във следващата популация с по-голяма вероятност следва да се включат индивиди, получени от индивиди от предната популация, които имат по-висока пригодност. За да се реализира това е необходимо от дадена популация да се селектират индивиди с висока пригодност (родители), от които да се получат нови индивиди (поколение).

2.3. Генетични операции.

Генетичните операции се използват за да се създаде поколение от родители. Основните генетични операции са генетично кръстосване и мутация. Генетичното кръстосване на двойка родители е илюстрирано чрез съответните хромозоми на фиг. 1.

↓

<i>00101101100101101110</i>	<i>00101101100011010101</i>
<i>11011101100011010101</i>	<i>11011101100101101110</i>

Фиг. 1. Генетично кръстосване

Мутацията е генетична операция, която се състои в изменението на един или няколко бита в даден хромозом. Мутациите се реализират в съответствие с предварително избрана вероятност. Позициите на мутациите се избират случайно. Мутации могат да се осъществяват както върху поколението, така и върху други индивиди, участващи в новата популация.

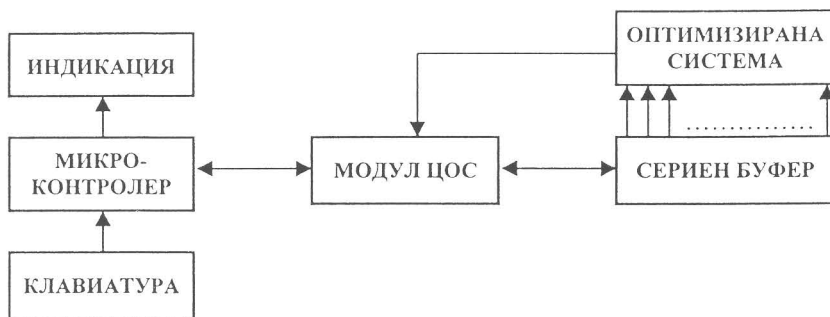
2.4. Формиране на нова популация.

В новата популация се включва поколението, част от индивидите от предната популация, които са с висока пригодност, неподложени на мутация и индивиди от предната популация, преминали мутация. За намаляване на вероят-

ността за захващане към локален екстремум е целесъобразно включването на определен брой случайни имигранти.

III. СТРУКТУРНА СХЕМА НА СИСТЕМА ЗА ГЕНЕТИЧНА ОПТИМИЗАЦИЯ В РЕАЛНО ВРЕМЕ

Към разработваната система се предявяват изисквания за високо бързодействие, универсалност на алгоритъма от гледна точка на приложимост при различни видове задачи. За да може системата да се използва за настройка на алгоритъма е целесъобразно операторът да има възможност да задава такива параметри на алгоритъма като размер на популацията, брой на параметрите в един хромозом, брой на битовете, кодиращи един параметър и др. За удовлетворяване на изискването за бързодействие е целесъобразно използването на процесор за цифрова обработка на сигнали (ЦОС), който реализира цялостно изчисленията, свързани с генетичния алгоритъм. За осъществяване на операторския контрол е предвидена подсистема, базирана на микроконтролер, който обслужва клавиатура, индикация и извършва обмен на параметри с модула за ЦОС. Структурната схема е дадена на фиг. 3.



Фиг. 2 Структурна схема на система за адаптация с генетичен алгоритъм.

Тази структура позволява на оператора да въведе параметрите на генетичния алгоритъм, които се наблюдават и на индикацията. Двупосочната връзка между микроконтролера и модула за ЦОС позволява от една страна предаване на въведените параметри към модула за ЦОС при подготовката на системата и предаване на данни за процеса на адаптацията от модула за ЦОС към микроконтролера и от там към индикацията. Разделянето на функциите между микроконтролера и модула за ЦОС е наложено от изискването за бързодействие, вследствие на което обслужването на по-бавните периферни устройства се осъществява от микроконтролера. По време на адаптацията модулет за ЦОС е ангажиран основно с изчисления по генетичния алгоритъм, като след всеки цикъл изпраща към микроконтролера данни за номера на цикъла и достигнатото качество. Чрез бърз сериен буфер модулет за ЦОС изпраща двоично

кодираните параметри към оптимизираната система и след изчакване на определено време за установяване нивото на изходният сигнал на системата, което се използва като критерий за качеството се въвежда в модула за ЦОС.

IV. ОСОБЕНОСТИ НА РЕАЛИЗАЦИЯТА НА ГЕНЕТИЧНИЯ АЛГОРИТЪМ ЧРЕЗ ЦИФРОВ СИГНАЛЕН ПРОЦЕСОР

Съвременните 16 и 32 битови цифрови сигнални процесори притежават достатъчно голяма памет за данните, което позволява реализиране на генетичния алгоритъм за оптимизиране на системи с голям брой параметри. Необходимият обем на паметта за съхраняване на хромозомите на популацията в битове се определя с израза $C = N \cdot M \cdot b$, където N е размера на популацията, M е броя на параметрите и b е броя на битовете, с които се кодира всеки параметър. Например при $N = 32$, $M = 64$ и $b = 8$ се получава необходим капацитет на паметта $C = 16384$ бита, което е напълно във възможностите на процесорите.

Реализацията на генетичния алгоритъм изисква генерирането на случайни числа при инициализирането и в хода на процеса както следва:

- а) При инициализирането на популацията - C двоични символа;
- б) При генетичното кръстосване - по едно случайно число за всяко кръстосване;
- в) При мутациите - по m случайни числа за определяне на наличието на мутация, където m е максималният брой мутации в един хромозом, и още по едно случайно число за позицията на всяка мутация;
- г) При създаването на хромозоми от типа случайни имигранти $C_{im} = N_{im} \cdot M \cdot b$, където N_{im} е броят на случайните имигранти в популацията.

От посочените най-голям е броят на случайните числа, необходими при създаването на хромозомите на случайните имигранти. Стандартният подход за генерирането на равновероятни независими двоични символи е генериране на случайно число X , равномерно разпределено в обхвата $0 \leq X < 1$ и се избере символ 0 при $X < 0.5$ и символ 1 при $X \geq 0.5$. Ускорено генериране на хромозомите на случайните имигранти може да се получи като се отчете, че генерираното случайно число в ЦСП е кодирано в двоичен вид с равномерно разпределени двоични символи и тези символи могат да се използват като символи на хромозома.

В разработения софтуер се реализира конгруентен метод [4], съгласно израза:

$$X_{n+1} = a \cdot X_n + b \pmod{T}.$$

Периодът на генератора T е избран 2^{32} , което позволява реализиране на достатъчно голям брой генетични цикли без повтаряне на случайните числа. Другите параметри са $a = 2^7 + 1 = 129$ и $b = 1$. Тези стойности позволяват бързо реализиране на генерирането, като умножението се заменя с една операция логическо отместване наляво с 7 бита и едно събиране.

На фиг. 3 е показана блоковата схема на алгоритъма за генетична адаптация, реализиран от модула за ЦОС. При инициализацията сигналният процесор



Фиг. 3 Блокова схема на алгоритъма.

приема параметрите на генетичния алгоритъм, зададени от микроконтролера и инициализира хромозомите на популацията като запълва съответната област от паметта със случайно генерирани двоични символи. В рамките на генетичния алгоритъм първо се извеждат към серийния буфер двоичните символи на текущия хромозом и се прочита изходният сигнал на физическата система. Това продължава докато се обработят всички хромозоми $n = N$. Следва сортиране на хромозомите според качеството. Ако не е достигнато зададеното ниво на качество алгоритъма продължава с изпращане на данни за номера на генетичния цикъл и достигнатото качество към микроконтролера и чрез него към индикацията. Извършва се отделно на най-добрите хромозоми и от тях се получава ново поколение чрез генетично кръстосване и мутации. Следва генериране на група от случайни имигранти и комплектоване на новата популация с което се подготвя новия генетичен цикъл. При достигане на зададеното ниво на качество двоичните символи на съответния хромозом се изпращат към серийния буфер, което осигурява работа на физическата система с най-добро качество. Модулът за ЦОС преминава в режим на контрол на изходния сигнал на системата и при изменение на външните условия и влошаване на качеството запуска отново генетичния алгоритъм за адаптация. Разработеният алгоритъм поради своята универсалност може да се разглежда като ядро на алгоритъм за генетична адаптация / оптимизация.

Операторът може да прекъсва процеса на адаптация чрез клавиатурата, при което микроконтролерът подава заявка за прекъсване към модула за ЦОС. При това се влиза в режим на задаване на параметри чрез микроконтролера.

V. ОСОБЕНОСТИ НА ПРАКТИЧЕСКАТА РЕАЛИЗАЦИЯ

В експерименталната система модулът за ЦОС се реализира с EZ-Kit Lite на Analog Devices, съдържащ 16 битов сигнален процесор ADSP2181, работещ с тактова честота 33 MHz. Процесорът разполага с 16к 24 битови думи памет за програмата и 16к 16 битови думи за данните. Най-значимите за генетичния алгоритъм операции се реализират с максимум 11296 цикъла (339 μ s) за сортирането и 17 цикъла (0.51 μ s) за генерирането на случайно число. Обменът с микроконтролера става посредством серийния интерфейс. Извеждането на данните към серийния буфер се осъществява чрез изводите FL0 и FL2 на процесора ADSP2181. Серийният буфер е реализиран с група CMOS двойни 4 битови изместващи регистри от типа 4015.

В управляващата подсистема се използва микроконтролерът PIC16F877, който е с тактова честота 4 MHz и разполага с 8к думи програмна FLASH памет и 368 байта памет за данни. Важни предимства на микроконтролера са от една страна големият брой входове / изходи (33), което позволява обслужване на клавиатурата, индикацията и обмена на данни с модула за ЦОС и от друга страна удобното програмиране поради наличието на FLASH памет. Използуваната индикация е буквено – цифров – цифров технокристален дисплей с 2 реда по 16 символа, позволяващ изобразяване на основните параметри на генетичния алгоритъм, номера на генетичния цикъл и достигнатото качество.

VI. ЗАКЛЮЧЕНИЕ

Предложените алгоритъм и система позволяват реализирането на генетичен алгоритъм за адаптация и оптимизация на физически системи с голям брой управляеми параметри в реално време. Основното предназначение на системата е адаптация на многоелементни антенни решетки, но поради универсалността на генетичния алгоритъм тя може да се използва при адаптация / оптимизация на други физически системи.

Това изследване е проведено в рамките на проект V РП – ТН – 1 / 1999 на Националния научен фонд.

ЛИТЕРАТУРА

1. K. S. Tang, K. F. Man, S. Kwong and Q. He, "Genetic algorithms and their applications", *IEEE Signal processing magazine*, vol. 13, No. 6, pp. 22-37, Nov. 1996.
2. R. Haupt, H. Southall, "Experimental adaptive nulling with a genetic algorithm", *Microwave journal*, Jan. 1999.
3. Foundations of Genetic Algorithms 2, Edited by L. Darrel Whitley, *Morgan Kaufmann Publishers, San Mateo, CA*, 1993.
4. Handbook of Mathematical Functions, Edited by Milton Abramowitz and Irene Stegun, *National Bureau of Standards*, 1961.