پنجمین کنفرانس سیستمهای فازی ایران
هفدهم الی نوزدهم شهریورماه ۱۳۸۳

بررسی روشهای تطبیقی فازی در الگوریتمهای ژنتیکی

ناهید پارسا، عادل رحمانی، ناصر مزینی، علی حمزه
دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت و ایران
parsa@iust.ac.ir

چکیده

الگوریتمهای ژنتیکی روشهای تکاملی کارآمدی در جستجوی فضاهای بزرگ و مسائل بهینه سازی هستند. در مدل کلاسیک الگوریتمهای ژنتیکی ، پارامترهایی نظیر احتمال تولید مثل و احتمال جهش ثابت در نظر گرفته شده اند ولی با توجه به تجربیاتی که تاکنون در زمینه الگوریتمهای ژنتیکی به دست آمده مشخص شده است که نیاز به کاوش[1] و بهره وری[2] در هر مرحله از اجرا، متغیر می باشد. سیستمهای فازی به علت قابلیت استفاده از دانش انسان و توانایی مقابله با عدم قطعیت، به خدمت الگوریتمهای ژنتیکی در آمده اند و کمک می کنند تجربیات مبهم ولی با اهمیت بدست آمده در زمینه کنترل الگوریتمهای ژنتیکی به صورت کاربردی مورد استفاده قرار گیرند. در این تحقیق روشهای تطبیقی فازی بر مبنای تطبیق پارامترها را مورد مطالعه قرار می دهیم. در این روشها یک کنترلر فازی با استفاده از یک سری قوانین فازی که تا کنون به صورت تجربی به دست آمده اند و یا توسط کنترلر کشف و فرا گرفته می شوند، با توجه به حالتی که الگوریتم ژنتیکی در آن قرار دارد، پارامترهای الگوریتم ژنتیکی را به صورت پویا به گونه ای تغییر می دهد که به کاوش و بهره وری مورد نیاز الگوریتم ژنتیکی دست یابد.

کلمات کلیدی: الگوریتم ژنتیکی- الگوریتم ژنتیکی تطبیقی- الگوریتم ژنتیکی فازی- تطبیق پارامترهای الگوریتم ژنتیکی

مقدمه

اگر چه فقط چند دهه ازپایه گذاری رایانش نرم[3] می گذرد، ولی بازده بالای این روشها در زمینه های مختلف هوش مصنوعی باعث افزایش کاربرد و سرعت گسترش آنها شده است. امروزه تمام ترکیبات شاخه های رایانش نرم (سیستمهای فازی، الگوریتمهای ژنتیکی، شبکه های عصبی و استدلال احتمالی) مورد استفاده قرار می گیرند.

هدف از این تحقیق نشان دادن توانایی بیشتر سیستمهای ژنتیک با پارامترهای تطبیقی نسبت به الگوریتمهای ژنتیکی سنتی که از پارامترهای ثابت استفاده می کنند، می باشد. الگوریتمهای ژنتیکی که در آنها سعی شده با استفاده از سیستمهای فازی، پارامترهای الگوریتم تغییر داده شده و بهینه سازی شوند تا بازده سیستم افزایش یابد.

پس از این مقدمه معرفی کوتاهی از الگوریتمهای ژنتیکی خواهیم داشت. سپس در مورد تطبیق[4] در الگوریتم ژنتیکی سخن گفته و ابعاد مختلف تطبیق را معرفی خواهیم کرد و به طور خاص مقدمه‌ای بر تطبیق با استفاده از سیستمهای فازی بیان می‌شود . بخش بعدی که عمده مطالب را در بر می‌گیرد، به معرفی چند روش ژنتیکی فازی می‌پردازد که در آنها جهت

1-Exploration
2-Exploitation
3-Soft Computing
4-Adaptation

بهینه‌سازی پارامترهای الگوریتم ژنتیکی از قبیل سایز جمعیت(N)، احتمال جهش(p_m) واحتمال تولید مثل (p_c) همزمان با تکامل پیدا کردن جمعیت از روشهای فازی استفاده شده است. پس از جمع بندی و خلاصه مطالب، در پایان فهرست منابع خواهد آمد.

الگوریتمهای ژنتیکی

الگوریتمهای ژنتیکی، روشهای جستجویی هستند که از تکامل طبیعی جهت یافتن جواب مسائل تقلید می کنند [1]. چنین الگوریتمهایی به جای اینکه در هر مرحله روی یک جواب کار کنند، روی تعدادی از جوابها (که یک جمعیت را تشکیل می‌دهند) کار می کنند. الگوریتم ژنتیکی یک مدل بر مبنای جمعیت است که از انتخاب، تولید مثل و جهش ژنتیکی استفاده می کند تا جوابهای جدیدی در فضای جستجو ایجاد کند. پیاده سازی یک الگوریتم ژنتیکی توسط ایجاد یک جمعیت از کروموزومها انجام می گیرد. سپس این کروموزومها ارزیابی می شوند و بر اساس اینکه چقدر"خوب" باشند(برازندگی[1])، به آنها اجازه تولید مثل داده می شود. "خوب" بودن یک کروموزوم به نسبت جمعیت کنونی سنجیده می شود[1]. یک الگوریتم ژنتیکی ساده می‌تواند به صورت زیر باشد:

1. Generate the initial population at random (P(0)).
2. REPEAT
3. Evaluate the fitness of each individual in P(i);
4. Select parents from P(i) based on their fitness in P(i);
5. ly search operators(Mutation and Crossover) to the parent and get generation P(i+1);
6. IL the population converges or the time is up.

تطبق در الگوریتم ژنتیکی

یک الگوریتم ژنتیکی شدیداً وابسته به میزان تعادل بین کاوش و بهره وری می باشد[2]. به این معنی که در یک الگوریتم ژنتیکی از جوابی که در حال حاضر بهترین است استفاده می کنیم (کاوش) و از طرفی سعی می کنیم جوابهایی را کشف کنیم که ممکن است از جواب حاضر بهتر باشد (بهره وری). در تمام مراحل باید مراقب بود که امکان تولید مثل را منحصر به بهترین کروموزومها نکنیم و به کروموزومهای دیگر نیز امکان تولید کروموزومهای جدید را بدهیم (عدم زیاده روی در بهره وری) از سوی دیگر نباید پیوسته به دنبال کروموزومهای جدید بود (عدم زیاده روی در کاوش)."انتخاب" بهترین‌ها و اهمیت ندادن به کروموزومها با برازندگی کمتر (فشار انتخاب)و از بین بردن ترکیبهای ژنی مناسب به هنگام تولید مثل (تخریب اسکیما)، باعث از دست دادن ژن های مناسب واز بین رفتن این تعادل و تنوع جمعیتی می شود. نتیجه این اتفاق‌ها، پر شدن محیط با جواب نسبتا خوب (همگرایی زودرس[2]) می باشد[2].

روشهایی برای کنترل رابطه ی بین بهره وری و کاوش[3] تا کنون ایجاد شده اند تا از همگرایی زودرس جلوگیری کرده و بازده الگوریتم ژنتیکی را افزایش دهند. این روشها در دسته بندی زیر قرار می گیرند[2]:

- پارامترهای تطبیقی(Adaptive Parameters): در این روشها پارامترهای الگوریتم ژنتیکی از قبیل سایز جمعیت(N)، احتمال جهش(p_m) واحتمال تولید مثل (p_c) با توجه به شرایط مسئله تطبیق داده می شوند.
- اپراتورهای تطبیقی(Adaptive Operators): در تطبیق اپراتورها با توجه به شرایط مساله، عملکرد اپراتورها (انتخاب، جهش و تولید مثل) تغییر پیدا کرده و تطبیق داده می شوند.
- انتخاب تطبیقی اپراتورها(Adaptive Operators Selection): در این دسته، از اپراتورهای مختلف در هر یک از سه مرحله انتخاب، تولید مثل و جهش استفاده می شودبه عنوان مثال در مرحله تولید مثل هم از n-point crossover [1] استفاده می شود و هم از Uniform crossover [1]. البته انتخاب نوع اپراتور بستگی یه شرایط الگوریتم دارد.

1-Fitness
2-Premature Convergence
3-Exploration and Exploitation Relation

- نمایش تطبیقی(Adaptive Representation): در این بخش، تطبیق را در شکل نمایش کروموزومها اعمال می کنیم.
- تابع برازندگی تطبیقی(Adaptive Fitness Function): تابع برازندگی نیز می تواند تطبیقی بوده و تعریف "میزان خوب بودن" یک کروموزوم با توجه به پارامترهای گوناگون، تغییر پیدا کند.

الگوریتمهای ژنتیکی تطبیقی فازی، از سیستمهای فازی در جهت بالا بردن بازده یک الگوریتم ژنتیکی استفاده می کنند. از آنجا که الگوریتمهای ژنتیکی علی رغم ظاهر ساده، ماهیت پیچیده ای دارند، غالب اطلاعاتی که در این زمینه تاکنون جمع آوری شده است، تجربی بوده و تابع قوانین منسجم که در قالب فرمول و روابط بگنجد، نمی باشند. از طرفی منطق فازی قادر است به خوبی با عدم قطعیت و ابهام مقابله کرده و دانش تجربی فرد خبره را وارد سیستمها کند[3]. در سیستمهای ژنتیک فازی که بر اساس تطبیق پارامترها عمل می کنند، پارامترهای الگوریتم ژنتیکی نظیر(p_m و p_c و N) توسط یک کنترلر فازی با توجه به شرایط الگوریتم در هر مرحله تطبیق داده می شوند. نحوه استفاده از کنترلر فازی، ورودی های کنترلر، پارامترهایی که تطبیق داده می شوند و ... در روشهای مختلف متفاوت هستند که در بخش بعدی به صورت جداگانه بررسی می شوند.

روشهای تطبیقی فازی

روش اول

در[4]و[5] "زو" و همکارانش برای رفع دو مساله اساسی در اجرای الگوریتمهای ژنتیکی از یک کنترلر فازی استفاده کرده اند . این دو مساله عبارتند از "سرعت پایین در جستجو" و "همگرایی زودرس". آنچه موجب ایجاد چنین مشکلاتی می شود می تواند یکی از سه مورد زیر باشد:

- ممکن است پارامترهای الگوریتم ژنتیکی(p_m و p_c و N)از ابتدا به خوبی انتخاب نشده باشند.
- ممکن است پارامترها به صورت ثابت[1] انتخاب شده باشند و با تغییر جمعیت، تغییری نکنند.
- اساسا انتخاب پارامترها به صورت تجربی به علت عملکرد پیچیده الگوریتمهای ژنتیکی بسیار مشکل می باشد.

الگوریتم "ژنتیکی فازی"[2] برای غلبه بر این مشکلات ارائه شد . این الگوریتم ترکیبی از ژنتیک و فازی بوده و به صورت زیر عمل می کند:

۱. پارامترها را قبل از آغاز عملیات انتخاب می کند.
۲. در طی اجرای الگوریتم ژنتیکی، به صورت همزمان پارامترها را کنترل کرده و به صورت پویا تغییر می دهد.
۳. در طراحی، پیاده سازی و تعیین مقدار تابع برازندگی کروموزومها به کاربر کمک می کند.

در این روش از دو کنترلر فازی برای تطبیق پارامترها استفاده می شود. ورودی هر دو کنترلر یکسان است و شامل نسل[3] و اندازه جمعیت می باشد.[4] خروجی کنترلر ها نیز p_m و p_c می باشند. فازی کنترلرها نیز از قوانین تجربی[5] که در جدول ۱و۲ آمده استفاده می نمایند.

این کنترلر فازی بر روی مسائل استاندارد GA از جمله مساله فروشنده دوره گرد، مورد بررسی قرار گرفت و نتایجی که به دست آمد از تمام روشهای معمول ژنتیکی بهتر بود.

1-Static
2-Fuzzy Genetic Algorithm
3-Generation
4-Population Size
5-Heuristic

جدول شماره ۱- جدول قوانین کنترلر فازی برای p_c

	اندازه جمعیت		
نسل	کم جمعیت	متوسط	پر جمعیت
اولیه	متوسط	کوچک	کوچک
میانی	بزرگ	بزرگ	متوسط
پایانی	خیلی بزرگ	خیلی بزرگ	بزرگ

جدول شماره ۲- جدول قوانین کنترلر فازی برای p_m

	اندازه جمعیت		
نسل	کم جمعیت	متوسط	پر جمعیت
اولیه	بزرگ	متوسط	کوچک
میانی	متوسط	کوچک	خیلی کوچک
پایانی	کوچک	خیلی کوچک	خیلی کوچک

o **روش دوم**

در [2]و[6] از یک کنترلر فازی استفاده شده که تشخیص می دهد در هر بارتولید مثل، احتیاج به کاوش ویا بهره وری داریم. این کنترلر فازی دو ورودی دارد:

- تنوع ژنتیکی ($d_{average}-d_{min}/d_{max}-d_{min}$) : ($d_{average}$، d_{max} و d_{min} به ترتیب عبارتند از میانگین، بیشترین و کمترین فاصله با بهترین کروموزوم.) در این الگوریتم قضاوت بر اساس وضعیت کروموزومها انجام می شود. تنوع ژنتیکی با میانگین فاصله ی اقلیدسی کروموزومها متناسب است. در صورتی که تنوع ژنتیکی نزدیک صفر باشد، *اکثر* کروموزومهای جمعیت در *نزدیکی* بهترین جواب جمع شده اند و به همگرایی *نزدیک* شده ایم. اگر تنوع ژنتیکی نزدیک یک باشد، کروموزومها در فضای جستجو پراکنده شده اند و تنوع جمعیت *زیاد* است.
- تنوع جمعیتی(فنوتایپی)($f_{best}/f_{average}$) : (f_{best}و$f_{average}$ به ترتیب عبارتند از میانگین و بهترین مقدار تابع برازندگی.) قضاوت بر اساس وضعیت جمعیت انجام می شود.در اینجا عکس پارامتر قبل، مقدار نزدیک به صفر نمایانگر تنوع جمعیت یا کاوش و مقدار نزدیک یک نمایانگر همگرایی می باشد.

خروجی این کنترلر فازی به شرح زیر است:

- میزان جستجو: یک مقدار حقیقی در محدوده ی [0-1] می باشد که *بزرگی* آن نیاز به کاوش و *کوچکی* آن نیاز به بهره وری را مشخص می کند. در صورت نیاز به کاوش از تولید مثلی که قابلیت تولید نقاط جدید در فضا دارد و در غیر این صورت از اپراتوری با قابلیت به کار گیری نقاط قدیم، استفاده می شود.
- فشار انتخاب: این پارامتر عبارت است از یک مقدار در محدوده ی [0-1] که میزان نیاز به فشار در مرحله انتخاب[1] را بیان می کند. هرچه به فشار انتخاب بیشتری نیاز باشد، میانگین برازندگی کروموزومهایی که به نسل بعد منتقل می شوند، بیشتر می شود.

1-Selection

جدول شماره۳- جدول قوانین کنترلر فازی برای میزان جستجو.

	تنوع جمعیتی		
تنوع ژنتیکی	کم	متوسط	زیاد
زیاد	کم	کم	متوسط
متوسط	زیاد	زیاد	متوسط
کم	زیاد	زیاد	متوسط

جدول شماره۴- جدول قوانین کنترلر فازی برای فشار انتخاب.

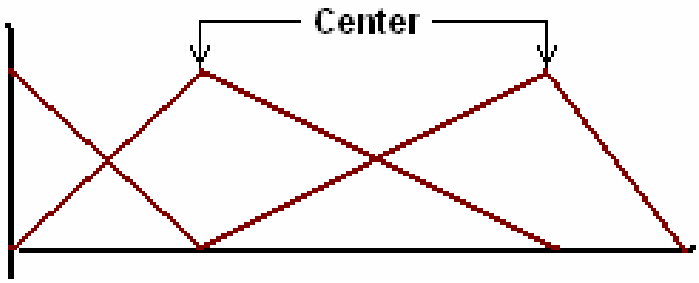
	تنوع جمعیتی		
تنوع ژنتیکی	کم	متوسط	زیاد
زیاد	کم	متوسط	زیاد
متوسط	کم	زیاد	زیاد
کم	کم	کم	زیاد

مجموعه قوانین[1] (جدول ۳و۴)و مجموعه های فازی این کنترلر به صورت تجربی به دست آمده است. البته در این روش از اپراتور و انتخاب تطبیقی نیز استفاده شده که در اینجا به آنها نپرداخته ایم.
نویسنده دو مقاله اخیر، در [7] روش خود را گسترش داده و برای ایجاد مجموعه قوانین و مجموعه های فازی از یک الگوریتم ژنتیکی استفاده کرده است.

- **روش سوم**

در[8]، "لی" یک سیستم الگوریتم ژنتیکی با پارامترهای پویا پیشنهاد کرده است. در اینجا یک کنترلر فازی وجود دارد که به صورت پویا پارامتر های الگوریتم ژنتیکی را تولید می کند. ورودی این کنترلر می تواند هرترکیبی از معیارهای بازده در الگوریتم ژنتیکی نظیر $f_{best}/f_{average}$ (f_{best}و$f_{average}$) به ترتیب عبارتند از میانگین و بهترین مقدار تابع برازندگی) باشد. برای مطالعه بیشتر این روش در ادامه به مشخصه های آن می پردازیم:

- شکل نمایش[2]: برای سهولت نمایش از توابع عضویت همجوار[3] استفاده شده است. در این توابع، راس هر تابع عضویت که مثلثی است، خود ابتدا و انتهای دو تابع عضویت دیگر که مثلثی هستند را مشخص می کند. بنا بر این برای نمایش n پارتیشن فازی، n-1 عدد که نمایانگر مراکز باشند، مورد نیاز است[8].



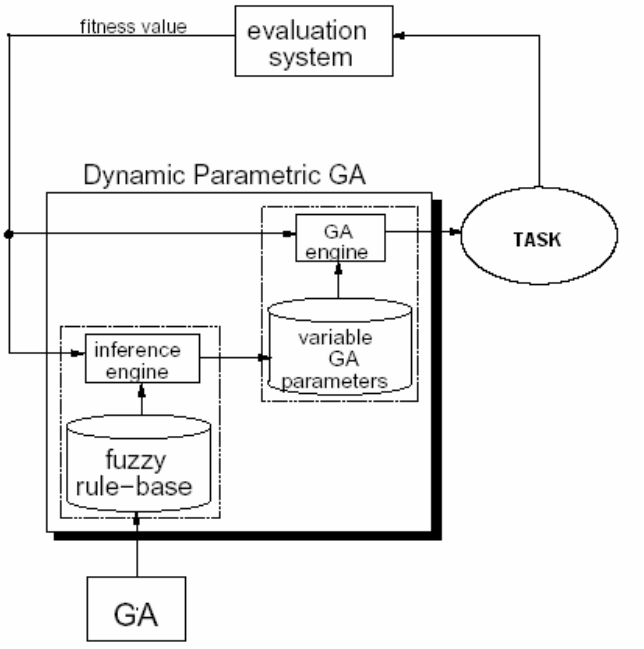
1-Rule base
2-Representation
3-Adjacent MBF

شکل ۱- مجموعه های فازی همجوار

- معیار بازده: این پارامتر متناسب با f_{best} انتخاب شده است.
- ورودی: می توان ورودی های متفاوتی در نظر گرفت اما معیارهای لی، $f_{bad}/f_{average}$ ، $f_{best}/f_{average}$ و توابعی از برازندگی بوده است.
- خروجی: این کنترلر فازی سه خروجی سایز جمعیت، تغییرات p_m و تغییرات p_c را تولید می کند.
- یادگیری: هدف الگوریتم ژنتیکی آموزش دهنده، آموزش یک مجموعه ی دانش[1] به کنترلر می باشد. خود مجموعه دانش از دو قسمت مجموعه قانون و تعریف پارامترها(که در اینجا همان مجمو عه های فازی هستند) تشکیل شده است. جمعیتی که توسط الگوریتم ژنتیکی آموزش دهنده تکامل می یابد، از کروموزمهایی که هرکدام یک مجموعه ی دانش برای فازی کنترلر هستند، تشکیل شده است.

یک مجموعه قانون می تواند به صورت زیر باشد:

IF (worst fitness)/(average fitness) IS small
THEN population size should decrease.
IF mutation IS small AND population size IS small
THEN population size should increase.



1-Knowledge base

شکل ۲- استفاده از الگوریتم ژنتیکی در طراحی اتوماتیک پایگاه دانش
یک فازی کنترلر در الگوریتم ژنتیک تطبیقی

نکته ای که وجود دارد این است که هر چه تعداد قانون ها کمتر شود، استحکام و کیفیت سیستم پایین می آید. چون از دقت سیستم کاسته می شود. در عین حال هر چه تعداد این قانون ها افزایش یابد، به علت صرف هزینه بیشتر در بررسی تعداد زیاد قوانین، بازده محاسباتی پایین می آید.

در سه روش بالا نحوه استفاده از کنترلر های فازی در تطبیق پارامترهای یک الگوریتم ژنتیکی را دیدیم؛ به این صورت که یک کنترلر یک سری ورودی از حالت کنونی سیستم می گیرد و با استفاده از پایگاه قوانین تعدادی خروجی کنترلی برای الگوریتم ژنتیکی در حال اجرا تولید می کند. یک مساله اساسی در استفاده از کنترلر فازی، پیدا کردن قوانین است. در طی عمر کوتاه الگوریتم های ژنتیکی، مجموعه ای از قوانین تجربی جمع آوری شده است که در دو روش اول، نمونه های آنها را دیدیم. اما اشکالی که وجود دارد این است که اولا این قوانین کامل نیستند و دوم اینکه برای حل تمام مسائل ژنتیک عمومیت ندارند. روش سوم برای مقابله با این مشکلات، به فراگیری قوانین توسط کنترلر فازی پرداخته است. این روش به علت جامعیت و تکیه نکردن به دانش تجربی اولیه، بسیار قدرتمند تر به نظر می رسد اگر چه بخش یادگیری که به سیستم اضافه شده ، هزینه اضافی بر سیستم تحمیل کرده است.

با کار بیشتر بر روی بخش یادگیری سیستم اخیرو افزودن پارامترهای ورودی بیشتراز الگوریتم ژنتیکی و یا حتی از خصوصیات مساله به کنترلر فازی، انتظار می رود به نتایج بهتری دست یابیم.

جمع بندی و ادامه کار

در طی این تحقیق، به معرفی سیستمهای ژنتیک با پارامترهای تطبیقی پرداختیم. در این سیستمها بر خلاف الگوریتمهای ژنتیک سنتی که از پارامترهای ثابت استفاده می کنند، پارامترها (N و p_m و p_c) با توجه به حالتی که سیستم در آن قرار دارد(نسل یا توابعی از برازندگی جمعیت)، تغییر می کنند. به طور خلاصه در تمام روشها از یک یا چند کنترلر فازی استفاده شده بود که پارامترها ی مختلفی که نمایانگر موقعیت کنونی الگوریتم ژنتیکی در رسیدن به جواب بود را به عنوان ورودی دریافت کرده و با استفاده از پایگاه داده ای که یا به صورت تجربی بدست آمده و یا توسط خود سیستم فرا گرفته شده بود، پارامترهای الگوریتم ژنتیکی را تولید می کردند. با جمع- آوری این قوانین در کنار هم، از تمام تجربیاتی که تا کنون به صورت منفرد به دست آمده می توان استفاده کرد و یک پایگاه قوانین منسجم برای تطبیق الگوریتهای ژنتیکی و کنترل آنها ایجاد کرد.

در گام بعدی یادگیری قوانین را نیز می توان به کار گرفت. در آخرین روش مورد بحث که از یادگیری استفاده کرده است، سیستم دیگر نیازی به دانش اولیه که تاکنون در زمینه ژنتیک جمع آوری شده است، ندارد و قادر است خود قوانین را استخراج کند و حتی به قوانینی جدید دست یابد. با ارزیابی ارائه شده در این تحقیق، به نظر می رسد روش تطبیقی فازی در بسیاری از موارد قادر باشد در جهت بالا بردن بازده یک الگوریتم ژنتیکی استفاده شود.

فهرست منابع

[1] Deb, K. "Genetic Algorithm in search and optimization: The technique and Applications." *Indian Institute Of Technology, Kampur.*
[2] Herrera, F, E. Lozano, M. (1996) "Adaptation of Genetic Algorithm Parameters based on Fuzzy Logic Controllers." *University of Granada, Spain.*
[3] Zimmerman, H.J. (1996) "Fuzzy Set Theory and its Applications." Third edition, *Kluwer Academic Publishers.*

[4] Xu, H. Y. Vukovich, G. (1993) "A Fuzzy Genetic Algorithm with effective Search Optimization." *Proc. Of 1993 International Joint Conference on Neural Networks.*
[5] H. Y. Xu, G. Vukovich, Y. Ichikava, Y. Ishii, (1994) "Fuzzy evolutionary algorithms and automatic robot trajectory generation." *Proc. Of the first IEEE Conference on Evolutionary Computation.*
[6] Herrera, F, E. Lozano, M. Verdegay, J, L. (1998) "Tackling Real-Coded Genetic Algorithms: Operators and tools for Behavioral Analysis." *University of Granada, Spain.*
[7] Herrera, F, E. Lozano, M. (1995) "Adaptive Genetic Algorithms based on Fuzzy Techniques." *University of Granada, Spain.*
[8] Lee, M, A. Takagi, H. (1993) "Dynamic control of Genetic Algorithms using Fuzzy Logic Techniques." *University Of California. USA.*