

# 一种改进的抑制早熟收敛的模糊遗传算法

邓莉 鲁瑞华

(西南大学电子信息工程学院信号与信息处理研究所 重庆 400715)

**摘要** 针对遗传算法中的早熟收敛现象,提出一种改进的模糊遗传算法。该算法将群体适应度均方差和种群的进化代数作为模糊逻辑控制器判断早熟收敛的标准,并根据判断结果对优劣不等的个体采取相应的进化方法,即当种群正常进化时对个体执行“惩强扶弱”的措施以保持种群多样性,一旦发生早熟收敛或有早熟收敛的趋势则对劣质个体进行局部突变,以恢复种群的进化能力。实验结果表明,与标准遗传算法、自适应遗传算法和模糊遗传算法相比,改进的模糊遗传算法能够更好地维持种群多样性,抑制早熟收敛。

**关键词** 早熟收敛,模糊遗传算法,进化代数,模糊逻辑控制器,种群多样性

## An Improved Fuzzy Genetic Algorithm to Suppress the Premature Convergence

DENG Li LU Rui-Hua

(Institute of Signal and Information Processing, School of Electronic Information Engineering, Southwest University, Chongqing 400715)

**Abstract** Aiming at the premature convergence of the genetic algorithm, an improved fuzzy genetic algorithm is proposed. In this algorithm, the mean square deviation of group fitness and population evolution generation are used as the criteria of premature convergence, and according to the estimation from fuzzy logic controllers, relevant evolution methods are given to different chromosomes, that is punishing the strongers and awarding the weakers when the population evolves normally in order to maintain the diversity of population, while doing catastrophe operation to the weakers to renew population evolution once the premature convergence appears or tends to appear. The experiment results show that the improved fuzzy genetic algorithm can maintain the population diversity and suppress the premature convergence better in comparison with the standard genetic algorithm, the adaptive genetic algorithm and the fuzzy genetic algorithm.

**Keywords** Premature convergence, Fuzzy genetic algorithm, Evolution generation, Fuzzy logic controller, Population diversity

## 1 引言

早熟收敛是遗传算法中的特有现象,它和算法收敛于局部最优不同<sup>[1]</sup>,在找到最优解或满意解之前,整个群体已收敛于一个随机的非优个体,而不一定是局部最优解,因此很难预见它的发生。早熟收敛的本质特征是群体中的个体非常相似,种群多样性急剧减少。当前群体缺乏有效的等位基因,在遗传算子的作用下不再生成高阶竞争模式。

导致种群早熟收敛的因素主要有群体规模、选择压力、适应度函数、初始群体分布等<sup>[2]</sup>,现有解决早熟收敛问题的方案<sup>[3]</sup>也主要针对以上各点。其中,基于模糊逻辑的遗传算法通过模糊规则调整交叉概率和变异概率,以维持种群多样性并加速收敛。在众多模糊规则中有一条重要原则<sup>[4~7]</sup>:当平均适应度与最佳适应度相差较大时,说明平均适应度相对较小,交叉概率和变异概率应取较大值。而文<sup>[8]</sup>则从理论上证明了种群早熟收敛的趋势与交叉概率的大小无关,交叉运算中的成熟化效应才是引起早熟收敛的主要原因,采用上述原则调整交叉、变异概率,只能起到加速搜索的作用,而不能真正抑制早熟收敛的发生。

另外,现有判断早熟收敛的方法主要是根据群体适应度均方差<sup>[9,10]</sup>在连续几代的变化情况来确定,若其变化微弱或保持不变,则说明种群早熟收敛。但仅靠这一变化量来断定

种群是否早熟收敛是不全面的。若在算法后期将全局收敛判断为早熟收敛而采取相应措施,无疑会对算法的稳定性造成冲击。

因此本文在传统模糊遗传算法的基础上,将群体适应度均方差和种群的进化代数一起作为模糊逻辑控制器(Fuzzy logic controller, FLC)判断早熟收敛的标准,并根据判断结果对不同的个体施以相应的交叉概率和变异概率,最后通过实验与其他几种算法进行比较,验证了本文算法在抑制早熟收敛和保持种群多样性方面的优势。

## 2 改进的模糊遗传算法

### 2.1 算法基本思想

将种群的进化代数作为判断早熟收敛的标准体现了一种时变思想<sup>[11]</sup>。种群多样性必将随着算法的收敛而逐渐减小,若在整个进化过程中都使用单一策略维持种群的多样性,则很可能影响算法的收敛速度,甚至出现误判而最终无法收敛。进化代数的功能与模拟退火算法中的温度  $T$  很相似,它们都宏观调整着算法的节奏与方向,但前者的实现方式却更加简明。模糊逻辑控制需要的只是“进化初期”、“进化后期”等模糊语言值,而不是像  $T$  那样细微缓慢的改变。

当模糊逻辑控制器判断种群正常进化时,算法对个体采取“惩强扶弱”的措施<sup>[12,13]</sup>,即优秀个体所得的交叉、变异概率

比它应得的偏低,劣质个体则偏高,这种趋势会随着进化代数  
的增加而逐渐减弱,以维持种群多样性并保证算法收敛;而当  
出现早熟收敛或有早熟收敛的趋势时,为保留群体已有的最优  
信息,算法只对低适应度个体采取高概率变异,若其进化产生  
适应度高于平均适应度的新个体,则用新个体取代它们本身。

### 2.2 模糊逻辑控制器设计

如图 1 所示,本文采用两个模糊逻辑控制器 FLC1 和  
FLC2 分别调整交叉概率  $P_c$  和变异概率  $P_m$ ,它们有两个相同的  
输入变量,即进化代数  $n$  和相邻两代群体适应度均方差的改  
变量  $\Delta E$ ,其中群体适应度均方差  $E$  的定义<sup>[9]</sup>如下:

$$E = \frac{\sum_{i=1}^N [f(x_i) - \bar{f}]^2}{N}, \bar{f} = \frac{\sum_{i=1}^N f(x_i)}{N} \quad (1)$$

式中  $N$  为最大遗传代数,  $f(x_i)$  为个体  $i$  的适应度,  $\bar{f}$  为群体  
平均适应度。  $\Delta E$  越小,表明群体适应度均方差的变化量越  
小;若  $\Delta E$  在进化初期或中期连续几代变化微弱或保持不变,  
则种群可能发生早熟收敛。为简化计算,将上式中的平方算  
子改为绝对值算子,表达式为:

$$E = \frac{\sum_{i=1}^N |f(x_i) - \bar{f}|}{N} \quad (2)$$

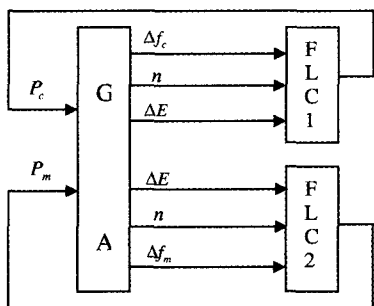


图 1 基于模糊逻辑控制器的算法框图

FLC1 的另一输入变量  $\Delta f_c$  为待交叉个体适应度较大者  
的适应度  $f'$  与最佳适应度  $f_{\max}$  之差,即

$$\Delta f_c = f_{\max} - f' \quad (3)$$

FLC2 的另一输入变量  $\Delta f_m$  为待变异个体的适应度  $f$  与  
最佳适应度之差,即

$$\Delta f_m = f_{\max} - f \quad (4)$$

$\Delta f_c$  和  $\Delta f_m$  越小,表明它们所对应的个体越好,反之则个  
体越差。

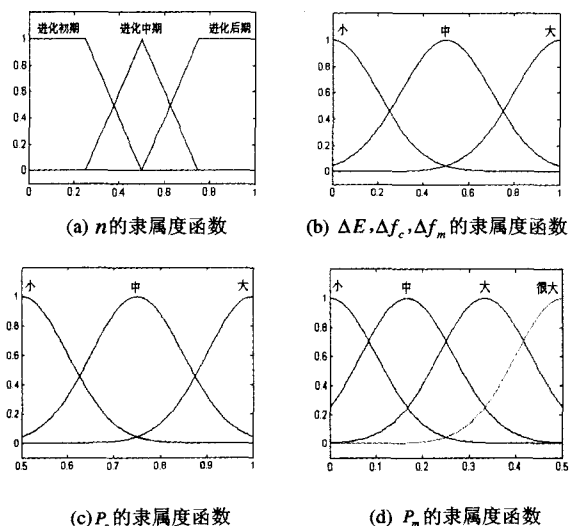


图 2 模糊逻辑控制器输入、输出变量的隶属度函数曲线

另外,各输入变量的模糊论域<sup>[14]</sup>都归一化到  $[0, 1]$  范围,  
输出变量  $P_c$  和  $P_m$  的论域分别为  $[0.5, 1]$  和  $[0, 0.5]$ 。输入  
变量  $n$  的隶属度函数取为梯形和三角形隶属度函数,定义分  
别如公式(5)和(6)所示。其中公式(5)的参数  $a, d$  分别对应  
梯形下部左右顶点,参数  $b, c$  分别对应梯形上部左右顶点;公  
式(6)的参数  $a, c$  分别对应三角形下部左右顶点,参数  $b$  对应  
三角形上部顶点。其余各变量的隶属度函数都取为高斯型隶  
属度函数,如公式(7)所示。其中  $\sigma$  决定函数中心点,  $c$  决定函  
数曲线的宽度<sup>[15]</sup>。各变量的隶属度函数曲线如图 2 所示。

$$f(x; a, b, c, d) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & d \leq x \end{cases} \quad (5)$$

$$f(x; a, b, c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (6)$$

$$f(x; \sigma, c) = e^{-\frac{(x-c)^2}{2\sigma^2}} \quad (7)$$

鉴于各模糊逻辑控制器的输入、输出变量较多,若变量的  
语言值也设置过多,模糊规则会相当复杂而难以实现。因此  
将各输入变量的语言值设为三个、输出变量  $P_c$  和  $P_m$  的语言  
值分别设为三个和四个,如图 2 所示,其中  $P_m$  的语言值“很  
大”对应于早熟收敛时局部灾变的高变异概率。这样,每个控  
制器就有 27 条模糊规则<sup>[16]</sup>,如表 1 所示。这些规则写成条  
件语句的形式为

if  $n$  is  $A_i$ , and  $\Delta E$  is  $B_i$ , and  $\Delta f_c$  is  $C_i$ , then  $P_c$  is  $E_i$ ;

if  $n$  is  $A_i$ , and  $\Delta E$  is  $B_i$ , and  $\Delta f_m$  is  $D_i$ , then  $P_m$  is  $F_i$ 。

其中  $i=1, 2, \dots, 27$ ,  $A_i, B_i, C_i, D_i, E_i, F_i$  是定义在  $n, \Delta E,$   
 $\Delta f_c, \Delta f_m, P_c, P_m$  论域上的模糊集。

在表 1 中,规则 1~3 和规则 10~12 分别表示进化初期  
和中期种群多样性变化微弱或停止改变,种群发生早熟收敛;  
而规则 7~9 表示进化初期种群多样性迅速下降,种群有早熟  
收敛的趋势。这两种情况都应对劣质个体实行局部灾变,即  
将  $\Delta f_c$  和  $\Delta f_m$  都为“大”的个体所对应的  $P_m$  取为“很大”。规  
则 4~6 表示进化初期种群正常进化,应对不同个体实行“惩  
强扶弱”的措施,即将  $\Delta f_c$  和  $\Delta f_m$  都为“小”的个体所对应的  
 $P_c$  和  $P_m$  取“小”,反之则取“大”,且措施应随进化代数  
的增加而减弱。当种群正常进化到中期,规则 13~15 将  $\Delta f_c$   
和  $\Delta f_m$  都为“大”的个体所对应的  $P_c$  和  $P_m$  由取“大”减小为  
“中”。规则 16~18 和规则 22~27 分别表示进化中期和后期  
种群多样性迅速下降,为保证种群快速收敛,将  $P_c$  和  $P_m$  随  
进化代数的增加由取“中”逐渐减小为“小”。规则 19~21 则  
表示进化后期种群多样性变化微弱或停止变化,种群已经收  
敛,只需维持  $P_c$  和  $P_m$  现有的语言值“小”至算法终止。

### 2.3 算法描述

根据上述分析,算法可由以下步骤描述:

Step1:生成初始种群。在规定取值范围内随机产生一定  
数量的个体。

Step2:编码。将个体编码为0,1二进制串。

Step3:适应度评价。将个体解码后代入目标函数求其适应度,并将个体按适应度值大小排序。

Step4:选择。采用随机遍历抽样方式从种群中选择一定数目的个体作为双亲繁殖后代。

Step5:用 FLC1 计算出一组交叉概率进行交叉操作。

Step6:用 FLC2 计算出一组变异概率,并查看该组变异概率的平均值是否超出某一阈值;若超过则说明种群个体在本代进化中的变异概率普遍偏高,种群有早熟收敛的趋势或已经早熟收敛,应对劣质个体进行局部灾变,其变异概率取阈值和该个体所对应的变异概率中较大的那个;若未超出则执行正常的变异操作。

Step7:以最大遗传代数作为结束条件,满足条件则终止算法,否则返回 Step3。

### 3 实验仿真

本文采用 De Jong's function 2 验证算法的性能。De Jong's function 2(简称 f2)又名 Rosenbrock's Valley,如图3所示,它的全局最优解位于一个抛物线形平坦且狭长的谷底。找到这条峡谷很容易,但要收敛到该最优解却很困难,因此出现早熟收敛的几率非常高。f<sub>2</sub>函数的定义如式(8)<sup>[17]</sup>所示,它的全局最优解为 f<sub>2</sub>(x)=0,与其相差 0.01 为满意最优解<sup>[2]</sup>。

表1 FLC1 和 FLC2 的模糊逻辑控制规则

R \ V	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
n	S	S	S	S	S	S	S	S	S	M	M	M	M	M	M	M	M	B	B	B	B	B	B	B	B	B	B
ΔE	S	S	S	M	M	M	B	B	B	S	S	S	M	M	M	B	B	B	S	S	S	M	M	M	B	B	B
Δf <sub>c</sub>	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B
Δf <sub>m</sub>	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B	S	M	B
P <sub>c</sub>	S	S	S	S	M	B	S	S	S	S	S	S	S	M	M	M	M	M	S	S	S	S	S	S	S	S	S
P <sub>m</sub>	M	M	H	S	M	B	M	M	H	M	M	H	S	M	M	M	M	M	S	S	S	S	S	S	S	S	S

注:表中的R表示各模糊逻辑控制器的规则,V表示输入或输出变量,表中每一列数据为该条规则所对应的各变量的语言值,其中对于变量n,S代表进化初期,M代表进化中期,B代表进化后期;对于其它变量,S代表小,M代表中,B代表大,H代表很大。

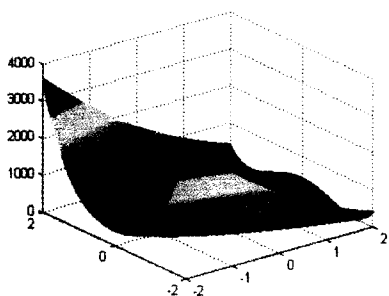


图3 二维 De Jong's function 2 图形

$$f_2(x) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2 \quad (-2.048 \leq x_i \leq 2.048, i=1,2) \quad (8)$$

实验选取标准遗传算法(Standard genetic algorithm, SGA)、自适应遗传算法(Adaptive genetic algorithm, AGA)、模糊遗传算法(Fuzzy genetic algorithm, FGA)和改进的模糊遗传算法(Improved fuzzy genetic algorithm, IFGA)进行比较,其中 SGA 的 P<sub>c</sub> 设为 0.7, P<sub>m</sub> 设为 0.01, AGA 确定 P<sub>c</sub> 和 P<sub>m</sub> 的数学表达式<sup>[7]</sup>如下:

$$P_c = \begin{cases} k_1(f_{max} - f') / (f_{max} - \bar{f}) & f' \geq \bar{f} \\ k_2 & f' < \bar{f} \end{cases} \quad (9)$$

$$P_m = \begin{cases} k_3(f_{max} - f) / (f_{max} - \bar{f}) & f \geq \bar{f} \\ k_4 & f < \bar{f} \end{cases}$$

设式中 k<sub>1</sub>=k<sub>2</sub>=1, k<sub>3</sub>=k<sub>4</sub>=0.5, f<sub>max</sub>, f', f,  $\bar{f}$  的定义与公式(2)~(4)中的定义相同。上述四种算法的群体规模设为 60, 个体的位串长度设为 10。

实验结果比较如图 4 和表 2 所示。图 4 为各算法在最大遗传代数为 100 时种群适应度平均值的变化曲线,表 2 为各

算法在最大遗传代数分别为 50、100、200 和 300 时重复计算 120 次求得平均优化结果。由图表可以看出,SGA 在进化初期种群多样性迅速下降,其早熟收敛比例平均高达 58.25%;AGA 的早熟收敛比例比 SGA 低 8.5%,收敛速度明显加快,在平均 28.375 代时种群就停止进化;FGA 则能很好地抑制早熟收敛,在最大遗传代数为 100 时种群几乎完全收敛,但由图 4 曲线可以看出其种群多样性变化缓慢,收敛非常迟钝;而 IFGA 能在进化前期保持种群多样性、抑制早熟收敛的同时在末期迅速收敛,并且它搜索到的满意解比 FGA 精确约 1.5963 倍,更接近全局最优解。

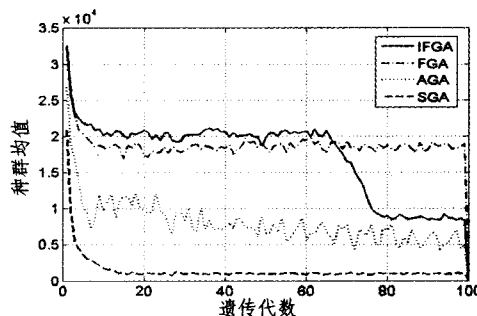


图4 种群适应度平均值变化曲线

结束语 由以上实验可知,改进的模糊遗传算法能够很好地保持种群多样性,抑制早熟收敛,并能较为精确地搜索到全局最优解。但是鉴于模糊逻辑控制器的输入、输出变量较多,模糊规则较为复杂,算法对于解决变量维数过多、精度要求很高的问题优势不明显,且算法在维持种群多样性的同时,其收敛速度还有待进一步提高。

表 2 四种算法对 f2 函数的优化结果

		SGA				AGA				FGA				IFGA			
		50	100	200	300	50	100	200	300	50	100	200	300	50	100	200	300
早熟收敛	W	0.0811	0.0656	0.0614	0.0628	0.0403	0.0450	0.0547	0.0451	0.0231	0.0101	—	—	0.0151	—	—	—
	T	24	47	116	174	13	20	15	14	43	89	—	—	43	—	—	—
	P	56%	63%	58%	56%	72%	46%	43%	38%	5%	0.8%	—	—	4%	—	—	—
全局收敛	W	0.0031	0.0028	0.0028	0.0020	0.0031	0.0028	0.0024	0.0024	0.0020	6.2521	2.2076	1.2893	0.0013	3.1480	1.3450	1.1429
	T	15	19	37	58	18	31	50	66	33	e-4	e-4	e-4	e-4	e-4	e-4	e-4
	P	44%	37%	42%	44%	53%	54%	57%	62%	95%	99.2%	100%	100%	96%	100%	100%	100%

注:表中的 W、T 分别代表早熟收敛或全局收敛时搜寻到的“最优解”和平均代数,P 代表早熟收敛或全局收敛所占的比例。

参考文献

- 雷英杰,张善文,李续武,等. MATLAB 遗传算法工具箱及应用. 西安:西安电子科技大学出版社,2005
- 李敏强,寇纪淞,林丹,等. 遗传算法的基本理论及应用. 北京:科学出版社,2002
- 王健,王建华. 标准遗传算法的研究进展. 华东船舶工业学院学报,2000,6(3):28~34
- 戚志东,朱新坚,朱伟兴. 基于模糊规则优化的改进模糊遗传算法. 计算机工程与应用,2003(27):18~20
- 李擎,郑德玲,唐勇,等. 一种新的模糊遗传算法. 北京科技大学学报,2001,2(1):85~89
- 张忠诚,王淳,李静,等. 基于模糊控制器的自适应遗传算法. 测控技术,2006(2):66~67
- Srinivas M, Patnaik L M. Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Trans Sys, Man and Cybern, 1994, 24 (4):656~667
- 徐宗本,高勇. 遗传算法过早收敛现象的特征分析及其预防. 中国科学(E辑),1996,8(4):364~375
- 严峻. 一种改进的基于成熟前收敛判断的自适应遗传算法. 南京邮电学院学报(自然科学版),1999,3(1):35~38
- 欧阳森,宋政湘,王建华,等. 一种快速收敛的遗传算法. 计算机应用研究,2003,9(3):50~52
- Last M, Eyal S. A fuzzy-based lifetime extension of genetic algorithms. Fuzzy Sets and Systems, 2005(49):131~147
- Andre J, Siarry P, Dognon T. An improvement of the standard genetic algorithm fighting premature convergence in continuous optimization. Advances in Engineering Software, 2001 (32):49~60
- 李旭东,涂葵生. 遗传算法中引入非自然规则的研究. 计算机工程与应用,2003,34:87~89
- 张筱磊,赵培庆. 基于模糊逻辑的遗传算法研究. 计算技术与自动化,2006,3(1):55~57
- 闻新,周露,李东江,等. MATLAB 模糊工具箱的分析及应用. 北京:科学出版社,2001
- 李国勇. 智能控制及其 MATLAB 实现. 北京:电子工业出版社,2005
- Strassner T, Busold M, Hermann W A. M M3 Parametrization of Four- and Five-Co-Ordinated Rhenium Complexes by a Genetic Algorithm. Journal of Computational Chemistry, 2002(23):282~290

(上接第 126 页)

表 2 系统响应类型的消息

消息类型	作用描述
LOGIN	协调模块向下载模块返回的注册成功与否的响应消息。
LOGOUT	协调模块向下载模块返回的注销成功与否的响应消息。
SEARCHCLIENT	协调模块返回给客户的拥有某个文件的客户的信息,返回结果为满足条件的客户的基本信息(id, ftpurl, ip, port)的列表。
VIEWCACHE	协调模块返回给客户的查询缓冲区情况的信息,返回结果为满足条件的客户的 id 列表。
RECEIVE	对等客户之间的下载响应消息,告诉对方它有这个文件,要求对方开始接收文件数据。
ADD_OK	协调模块向客户返回的已经成功增加一个文件记录的消息。
DELETE_OK	协调模块向客户返回的已经成功删除一个文件记录的消息。
SERVER_STOP	协调模块自己退出时,向所有客户发送退出信息。
NOFILE	对等客户之间请求下载文件时,如果文件不存在则给对方返回 NOFILE

3.4 缓冲区的管理方法

这个系统的主要特征是利用网络上各个计算机的缓冲区,形成一个巨量的丰富的可共享的资源。每个客户都划出一块区域作为缓冲区,用于存放共享的文件。这就要求对下载模块的缓冲区做出合理科学的管理与组织。

客户缓冲区中所出现的文件用该文件的 ftpurl 来标志,即此文件在原来的 Ftp 服务器上的 URL。例如,一个文件在 Ftpserver 服务器上的路径为/dir1/dir2/filename,则该文件在客户机的存放路径就是 cacheroot/ftpserver/dir1/dir2/filename,其中 cacheroot 为客户缓冲区的路径。这里称 Ftpserver/dir1/dir2/filename 为该文件的 ftpurl,这个 ftpurl 记录了文件的基本信息。

客户缓冲区内的文件数目有一个上限。当达到上限时,就要淘汰其中一些文件,再通知协调模块更新,这样新文件就

可以加入到缓冲区。本系统采用 LRU 淘汰算法,即选择一个最近最久未使用的文件进行淘汰。因此,下载模块中,需要用到两个向量 fileinfo 与 lastusetime,分别记录缓冲区每个文件的 ftpurl 与最后一次的使用时间。当需要淘汰一个文件时,下载模块从向量 lastusetime 中找出时间最小的一项,获得对应的索引,然后在那两个向量中删除对应的项,并删除相应的缓冲区中的文件。最后,下载模块再通知协调模块更新该客户相应的文件记录信息。

**总结与将来的工作** 该系统在局域网内能正常运行,性能良好。当然,系统还有一些方面需要改进与优化,以便扩充与增强系统的性能。主要计划中的改进包括:分段下载/缓存功能,对私有网段间的客户端通信的支持。我们将使用下列方法来做出这些改进。

进行分段下载/缓存时,客户首先搜索到拥有某个文件的其它客户列表。根据一定的算法,选择从列表中不同的客户那里下载文件的不同的块。这样做可以有效提高下载速度。

如果两个客户处于不同的私有网段内,因为它们的 IP 不具有唯一性,所以它们不能直接通信。要实现它们之间的通信,就要使用 NAT 穿越技术。NAT 穿越涉及 TCP/IP 网络中的一个常见问题,即在处于使用了 NAT 设备的私有 TCP/IP 网络中的主机之间建立连接的问题。我们将使用 UDP hole punching 技术进行 NAT 穿越<sup>[6]</sup>。

参考文献

- 网络软件 Ftp 工具. LeapFTP 软件介绍. http://www.sky-cn.com/soft/710.html
- Ftp 工具. FlashGet 的特点简述. http://zhidao.baidu.com/question/8627776.html
- CuteFTP. V6.0.3 Home 功能描述. http://www.qylt.com/download/soft/346.htm
- FlashFXP 软件的特点. http://soft.cnzz.cn/soft/html/43.html
- 常用 BT 软件介绍. BitTorrent. http://zhidao.baidu.com/question/4488755.html
- Zhou Hu. NAT Traversal Techniques and Peer-to-Peer Applications. Telecommunications Software and Multimedia Laboratory, Helsinki University of Technology, Oct. 2005