

# Cognitive Engine of Case-Based Reasoning Based on VC ++ 6.0 and SQL

Guoping Wu, Lei Shen, Yuan Liu, Luping Zhang

School of Communications Engineering, Hangzhou Dianzi University, Hangzhou  
Email: shenlei@hdu.edu.cn

Received: Sep. 9<sup>th</sup>, 2013; revised: Sep. 16<sup>th</sup>, 2013; accepted: Sep. 18<sup>th</sup>, 2013

Copyright © 2013 Guoping Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Abstract:** This paper aims to use the cases of SQL 2008 database storage, multi-objective optimization results and economic benefit of functions, to solve the application problems of dynamic resource allocation based on case-based reasoning with optimization process displayed dynamically in MFC interface. VC++6.0 visit SQL2008 database by ADO database access technology, go through the date in case database, choose the execution plan most matching with environmental database, implement the optimization algorithm, sent the optimized parameters configuration to database output module, and finally update case database by optimized parameters and utility value. Practice shows that this scheme is effective because it can select optimization algorithm dynamically, conduct case update and optimization procedure and it leads to the dynamic resource allocation of cognitive radio.

**Keywords:** VC++; SQL 2008; Case-Based Reasoning; Cognitive Radio

## 基于 VC++6.0 和数据库的案例推理认知引擎

武国平, 沈雷, 刘源, 张璐苹

杭州电子科技大学通信工程学院, 杭州  
Email: shenlei@hdu.edu.cn

收稿日期: 2013年9月9日; 修回日期: 2013年9月16日; 录用日期: 2013年9月18日

**摘要:** 针对基于案例推理的动态资源分配的实际应用问题, 本文利用 SQL2008 数据库存储案例、多目标优化结果以及效益函数, 在 MFC 界面上动态显示优化过程。VC++6.0 选用 ADO 数据库访问技术访问 SQL2008 数据库, 对案例子库的数据进行遍历, 选择与环境子库中最为匹配的执行方案, 执行优化算法, 最后将优化后的参数配置送至数据库输出模块, 同效用值一起更新案例子库。实践证明本方案灵活有效, 可以动态选择优化算法, 进行案例更新、优化过程, 成功地实现了认知无线电资源动态分配的问题。

**关键词:** VC++; SQL 2008; 案例推理; 认识无线电

### 1. 引言

认知无线电(CR)技术<sup>[1]</sup>是解决频谱资源匮乏的有效手段, 近几年来, 对频谱分配的研究非常广泛, 而基于案例推理(case-based reasoning CBR)<sup>[2]</sup>的动态资源分配技术引起了广泛的关注。贪婪算法<sup>[3]</sup>、背包算

法等需要搜索整个解空间, 然后通过比较选择出最优的参数配置, 不能满足认知无线电系统实时性的要求。案例推理技术是一种人工智能学习方法, 模仿人类基于经验或案例来进行学习、理解、处理问题, 可以通过把现有无线电环境信息与历史环境信息相比

较，寻找最相似的案例，利用先前的执行方案指导处理现在的问题。但是当历史方案应用于现有问题的效用值不高，即与当前环境信息相似度不高时，我们需要对历史执行方案进行适当的调整，以满足当前环境的传输要求。在本文中，我们可以利用量子遗传和粒子群算法实现参数配置的优化过程。DS-CDMA 在 GPS 系统，低轨道卫星通信系统，无线传感网以及 3G 网络等军用民用系统中都有比较广泛的应用。

量子遗传算法<sup>[4]</sup>和粒子群算法<sup>[5]</sup>都包括了染色体编码、种群初始化和种群更新过程，理论简单，具有寻优能力强、早期收敛速度快、收敛后稳定性强以及搜索时间短等优点。

关于认知无线电动态资源分配的理论研究非常多，都是基于 Matlab 给出仿真结果，而在软件实现这块一直存在空白。本文利用 SQL2008<sup>[6]</sup>数据库存储案例、多目标优化结果以及效益函数，在 MFC<sup>[7]</sup>界面上动态显示了优化过程，软件实现了发射功率、调整方式的参数调整。

## 2. 软件工作流程

根据项目要求，本软件由数据输入模块、数据输出、多目标优化以及案例推理四个基本模块组成，根据组成模块之间的相互关系，软件工作流程图如图 1

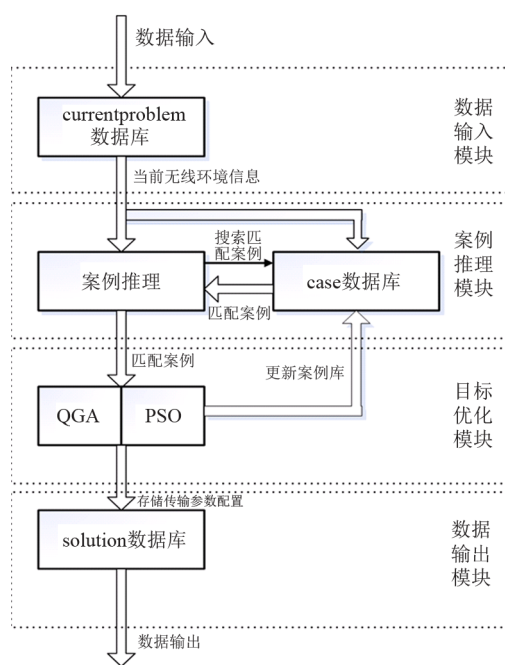


Figure 1. Flow chart of software work  
图 1. 软件工作流程图

所示。

在 SQL2008 中，我们新建了 3 个数据库，分别为 currentproblem 数据库、case 数据库以及 solution 数据库。在 case 数据库中，包含了 3 个表：h 表(存放信道衰落因子)，power 表(存放发射功率)和 mod 表(存放调制方式)，solution 数据库中存放表 outcome，各个表中的连接通过相同的主键 CaseId 实现互联。

数据输入模块将外界无线环境信息读入到本地数据库，即 currentproblem 数据库中，供多目标优化模块和案例推理模块调用；数据输出模块则是将优化结果存放到 solution 数据库中，方便管理者直观读取结果；多目标优化模块主要是根据目标函数进行优化，以适应外界无线环境信息及用户需求，功能在 VC++6.0 中实现；案例推理模块包含了在 case 数据库中进行案例比较、案例查找与选择、案例更新。

## 3. 软件功能分析

### 3.1. 数据库连接

数据输入模块和数据输出模块都涉及到了对数据库的数据进行读写的操作。目前，VC 访问 SQL Server 数据库，常用的方法有两种：通过 ODBC 连接和通过 ADO 连接。

采用 ODBC 方式要求先创建 ODBC 数据源，应用程序通过数据源与数据库连接；并且在创建应用程序时，只能是基于单文档或多文档，而无法直接创建基于对话框的应用程序。而采用 ADO 方式无需创建数据源，应用程序可通过 ADO 直接与数据库连接；ADO 方式可以创建基于对话框的应用程序。因此我们选择了 ADO 方式。

我们通过 COM 智能指针 \_ConnectionPtr m\_pConnection 和 \_RecordsetPtr m\_pRecordset 调用 COM 组件的方法，实现数据库操作的基本功能。

ADO 封装类里面 CADODatabase 类与 CADORRecordset 类，CADODatabase 类主要封装了连接数据库的功能，CADORRecordset 类主要封装了对数据库记录操作的功能。

在 CADODatabase 类中，OpenDB(LPCTSTR lpstrConnection)函数实现打开数据库连接的功能，IsOpen()以及 Close 函数实现链接的打开判断及关闭操作，错误处理及错误描述通过 dump\_com\_error(\_com\_er-

ror&e)函数实现。在 CADORRecordset 类中, Open(LPCTSTR lpstrExec, int nOption)、IsOpen()、Close()函数实现记录集的连接、打开判断以及关闭。记录集字段的获取通过 SetFieldValue()函数实现, 其中, 形参的不同表示类型用于获取不同种类的数据。记录集遍历操作通过 MoveFirst()、MoveLast()、MoveNext()等函数实现。我们用 AddNew()、Delete()、Update()函数达到添加/删除单个记录、更新记录集的目的。同样的, 错误处理及错误描述通过 dump\_com\_error(\_com\_error&e)函数实现。

### 3.2. 多参数调整

多目标优化模块的功能是对匹配案例的执行方案进行调整优化, 以满足当前用户和网络的需求, 最后将优化后的参数传输至数据输出模块和案例推理模块。而这个调整优化过程, 就需要涉及到多参数调整函数, 本文采用的优化算法为粒子群算法和量子遗传算法, 在实验界面中, 对两种算法选其一, 进行发射功率和调制方式的参数优化。以下就对选择粒子群算法的原理以及 VC++实现进行简单介绍:

#### 3.2.1. 粒子群算法原理

粒子群算法源于对鸟类捕食行为的研究, 通过交流同伴位置信息和历史最好位置信息来更新当前位置, 是一种全局搜索最优解算法。假设粒子群规模为  $S$ , 则在一个  $D$  维的搜索空间中, 粒子  $i$  的位置信息可以表示为  $x_i^k = [x_{i1}^k, x_{i2}^k, \dots, x_{iD}^k]$ , 速度信息表示为  $v_i^k = [v_{i1}^k, v_{i2}^k, \dots, v_{iD}^k]$ , 粒子  $i$  到目前所经历的最好位置  $p_i^k = [p_{i1}^k, p_{i2}^k, \dots, p_{iD}^k]$ , 其中  $k$  为第  $k$  次迭代, 则群体中所有粒子到第  $k$  次迭代结束后经历过的最好位置为  $p_b^k = [p_{b1}^k, p_{b2}^k, \dots, p_{bD}^k]$ , 其中  $b$  为具有最优位置粒子的索引。在本文算法中,  $D$  是 2 维搜索空间, 分别是发射功率和调制模式, 最好位置指的是适应度最高的解。

在迭代过程中, 速度的更新方程为:

$$v_{id}^k = v_{id}^{k-1} + c_1 r_1 (p_{id}^{k-1} - x_{id}^{k-1}) + c_2 r_2 (p_{bd}^{k-1} - x_{id}^{k-1}) \quad (1)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (2)$$

其中  $c_1$  和  $c_2$  为学习因子,  $r_1$  和  $r_2$  是分布于  $[0,1]$  之间的随机数。为了避免粒子离开搜索空间,  $x_{id}^k$  被限制在  $-V_{max}$  和  $V_{max}$  之间。

本文采用二进制粒子群算法, 因此位置信息  $x_{id}^k$  和  $p_{id}^k$  都被限定为 0 或 1, 速度大小需要通过下式转化到  $[0,1]$  之间的概率值:

$$S(v_{id}^k) = \frac{1}{1 + \exp(-v_{id}^k)} \quad (3)$$

二进制粒子群算法的粒子位置更新公式为:

$$x_{id}^k = \begin{cases} 1, r < S(v_{id}^k) \\ 0, r \geq S(v_{id}^k) \end{cases} \quad (4)$$

其中  $r$  为在  $[0,1]$  之间均匀分布的随机数。

#### 3.2.2. 算法软件实现

由 2.1 介绍可知, 本文粒子群算法是基于二进制型的, 因此在初始化的时候, 必须要把案例库中获取的匹配案例的发射功率和调制方式转化成二进制型, 这里, 我们通过 multicode 函数实现多载波编码。每个子载波的发射功率范围为 0~25.2, 精度为 0.4, 编码为 6 bit, 调制方式可为 1, 2, 3, 4, 因此编码为 2 bit, 单载波编码通过函数 code 实现, 我们假设有 10 个传输信道, 因此多载波编码好了的数据为一个大小为 80 的数组  $(8*10)$ , 存放在 bestp[80] 中。速度的初始化由 C++ 自行产生 0~1 的随机数得到。

每次进化(迭代)运算后, 需要得到效用函数值, 根据效用函数的大小, 分别在 pimax[population][80] 和 pgmax[80] 存放粒子  $i$  到目前所经历的最好位置及所有粒子到目前所经历的最好位置, 其中 population 为种群规模。效用值大小通过函数 fitness 实现。然后应用公式(1), 更新速度信息, 根据公式(4), 更新位置信息  $x_{id}^k$ 。

Particles warm optimization 函数调用以上函数, 实现进化, 完成后返回效用值 fgmax、多载波发射功率以及多载波调制方式。

### 3.3. 案例比较、更新、匹配案例选择函数

案例比较函数是比较数据输入模块中的数据, 即 currentproblem 数据库中的数据同 case 库中 h 表中的信息进行比较, 函数定义为 Evf (double \*dblvalue1, double \*dblvalue2), 函数返回相似值, 相似值越高, 代表案例库中的当前比较案例同当前环境描述越接近, 因此我们可以通过遍历案例库中的案例, 根据相

似度函数的返回值选取匹配案例，返回该案例的索引号，该功能通过匹配案例选择函数 SimilarCase()实现。案例库的大小有一定的规模限制，太小，不容易找到特别近似的案例，延长优化算法的执行时间，案例库规模太大，则由于遍历算法的存在，延长匹配案例索引号的获取时间。因此，案例库规模的选择经过衡量，本文采用 150 个案例的规模。初始条件下，案例库为空，直接执行优化算法，把信道条件和参数配置作为新案例添加到 case 库中，案例库中标号+1，所要添加数据的表格为所有表格，即表 h、power、mod。当案例库饱和后(本案例库案例个数上限预定为 150 个)，查找案例库中系统效用值最小的案例(通过在函数中

调用 ADO 类，在 outcome 表中查找)，用当前案例替换，存放到案例库中。所要替换掉的内容包括表 h、power、mod、outcome 对应序号所存放的数据。案例更新所涉及到的功能通过 UpdateDB()函数实现。

#### 4. 软件实现界面

如图 2 所示，这是基于案例推理的动态资源分配主界面。当我们选择 CBR-PSO 算法，目标模式选择为 mod1 时，我们按“确定”按钮，程序开始执行，我们可以从主界面上直观看到程序运行过程，读取优化结果。我们也可通过数据库查询按钮查询 case 库和 solution 库中信息。如图 3 所示，为点击“案例库数

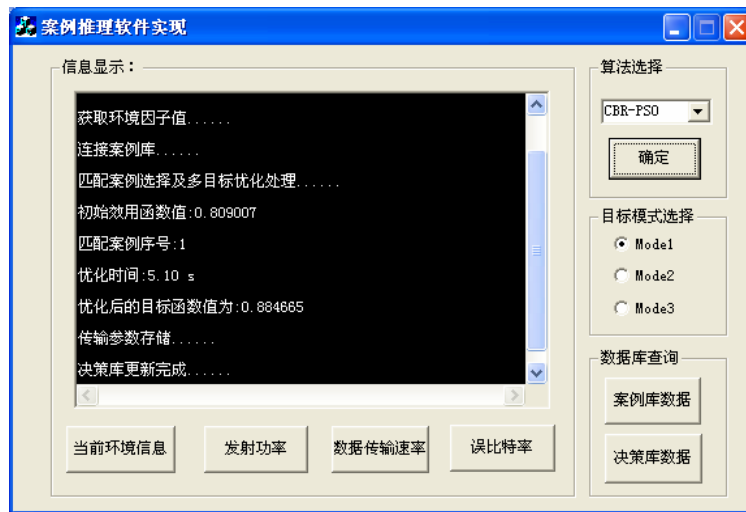


Figure 2. Main interface of dynamic resource allocation  
图 2. 动态资源分配主界面

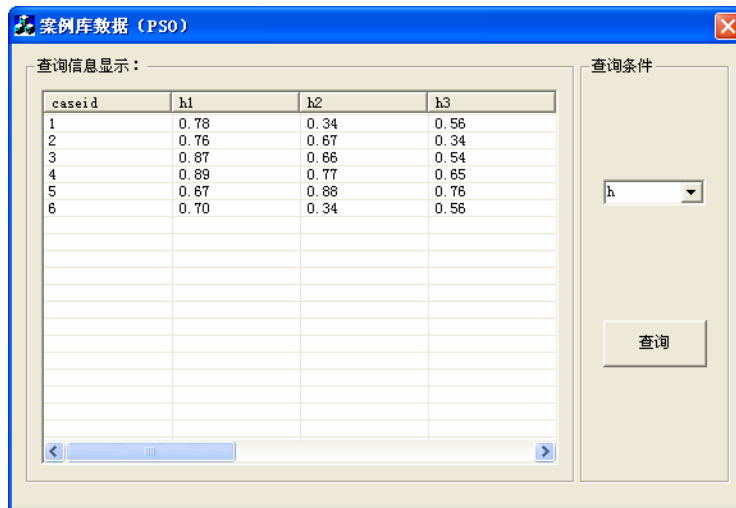


Figure 3. Date query of case base  
图 3. 案例库数据查询

据”按钮后弹出的窗口，可以在查询条件的下拉菜单中选择需要查询的表格，进行查询操作。

## 5. 总结

本文利用 SQL2008 数据库存储案例、多目标优化结果以及效益函数，在 MFC 界面上动态显示优化过程。弥补了在认知无线电在软件实现方面的空白，具有实际应用价值。在算法方面，CBR-PSO 和 CBR-QGA 的优势在于，在优化算法执行之前，抛弃了那些不可能产生更好性能的配置，把搜索空间缩小到最优解的周围，加快了算法的收敛速度。通过编程运行证明，本方案灵活有效，可以动态选择优化算法，进行案例更新、优化过程，成功地实现了认知无线电资源动态分配的问题。

## 6. 致谢

本论文得到浙江省大学生新苗计划项目的资助，

在此表示感谢。

## 参考文献 (References)

- [1] Federal Communications Commission. Spectrum policy task force report. ET Docket, 2002, 02-135.
- [2] R. C. Schank. Dynamic memory: A theory of reminding and learning in computers and people. New York: Cambridge University Press, 1982.
- [3] D. Sun, B. Zheng. A novel multi-user low complexity bit allocation algorithm in cognitive OFDM networks based on AM-GM inequality. IET 3rd International Conference on Wireless, Mobile and Multimedia Networks (ICWMNN 2010), 2010: 217-220.
- [4] A. Narayanan, M. Moore. Quantum-inspired genetic algorithms. Proceedings of IEEE International Conference on Evolutionary Computation, 1996: 61-66.
- [5] J. Kennedy, R. C. Eberhart. A discrete binary version of the particle swarm algorithm. IEEE International Conference on Evolutionary Computation, 1998: 84-88.
- [6] 康会光, 马海军. SQL Server 2008 中文版标准教程[M]. 北京: 清华大学出版社, 2009.
- [7] S. B. Lippman, J. La Joie and B. E. Moo. C++Primer[M]. 北京: 人民邮电出版社, 2006.