

全自动酶免工作站计算机控制系统设计

陈珂², 徐科^{1,3}(1. 苏州大学 信息处理及应用研究所, 江苏 苏州 215006; 2. 苏州市职业大学, 江苏 苏州 215004;
3. 苏州幻方信息科技有限公司, 江苏 苏州 215000)

摘要:介绍了酶免工作站软件控制部分的分析和设计。按软件工程的设计思想, 归纳出系统的对象, 给出系统实现单文档多视和控制类的详细设计。并将一种改进的自适应遗传算法应用于系统执行效率的优化。此开发系统已投入实际试验, 从实际使用的效果看, 试验执行效率提高了69.9%, 血样检测的准确率也达到了99.6%以上, 因此, 每台酶免仪每天可以多检测120个左右的血样, 如果按照现在医院血液检测的指导价, 每台酶免仪每年可为医院多创造150万人民币左右的利润。

关键词:酶免仪; 计算机控制; 遗传算法; 自适应

中图分类号: TP18

文献标识码: A

文章编号: 1673-629X(2008)06-0160-04

Design of Computer Control System for Automated
ELISA WorkstationCHEN Ke², XU Ke^{1,3}(1. Institute of Intelligent Information Processing and Application, Soochow University, Suzhou 215006, China;
2. Suzhou Vocational University, Suzhou 215004, China;
3. Suzhou Cube Information Technology Co., Ltd, Suzhou 215000, China)

Abstract: Puts forward the design and analysis of the control system. Follow the thinking of software engineering, conclude systems' objects, describe properties. At last, give the exact design of control class and interface, and apply a modified adaptive genetic algorithm to the optimization of system. The system has been experimented, the result shows our workstation has increased 69.9% efficiency, and the accuracy also reached more than 99.6%. Therefore, each workstation can detect more blood samples of about 120, and each enzyme immunoassay instrument can create more than 1.5 million yuan of profits for the hospital.

Key words: ELISA workstation; computer control; genetic algorithm; adaptive

0 引言

1971年瑞典学者 Engvall 和 Perlmann, 荷兰学者 Van Weerman 和 Schuurs 分别报道将免疫技术发展为检测体液中微量物质的固相免疫测定方法(称为酶联免疫吸附试验)。原理是使酶与抗体或抗原结合, 用以检查组织中相应的抗原或抗体的存在。后来发展为将抗原或抗体吸附于固相载体, 在载体上进行免疫酶染色, 底物显色后用肉眼或分光光度计判定结果。这种技术就是目前应用最广的酶联免疫吸附试验, 俗称 ELISA^[1]。

由于 ELISA 步骤繁多, 手工操作中任何一个疏

忽, 都会使检验结果的可靠性大受影响。近几年, 一些国外大公司推出了全自动酶免分析系统, 它们完全替代了手工操作, 极大地提高了检测结果的可靠性。国内对于免疫酶测定方法的研究相对起步较晚, 免疫酶测定方法研究的配套仪器国产化也落后于国外。随着国内医疗卫生水平的提高, 以及各种病毒检测的普及, 全自动酶免设备的需求将不断扩大, 而现在国内销售的酶免仪产品, 全部为国外的产品, 价格昂贵。针对这些问题, 自主研发了全自动酶免工作站, 系统结构及功能如下:

1) 系统结构。整个酶免工作站由机械部分和控制部分组成。机械部分被封装在一个两层的机架内。机架的上层放置配液模块, 包括两个机械臂: 一个用来吸液, 吐液, 取针, 脱针, 该机械臂上装有一个压力传感器, 用来测页面压力; 另一个机械臂用来取放培养板;

收稿日期: 2007-09-16

基金项目: 江苏省科技发展计划 (BS2002015)

作者简介: 陈珂 (1974-), 女, 硕士, 讲师, 主要研究方向为信息安全、图像处理。

一个条码扫描枪,用来扫描每个病人的id号;另附有针头座、试剂座、血样座、拖链等。下层放置孵育、读板、洗板模块,主要包括孵育仓,读板机,洗板机;另附有导轨。孵育仓内有8个孵育位,每个孵育位有1只温度传感器,系统可单独控制每个孵育位的温度。机架外侧有一台PC作为控制主机,可以和医院病人数据库进行数据交换。

2)功能描述。(1)数据采集 系统每隔10秒对孵育仓每个孵育位的温度进行采集,并做抗干扰滤波处理,得到孵育仓内每个孵育位的温度参数,作为控制的依据;机械臂每次在进行吸液操作之前,都必须根据压力传感器返回的压力值来判断针头是否到达液面以下一定的深度;(2)试验配置 设置试验中每次吸液、吐液的量,孵育的时间、温度,实验中要使用到的药品的摆放位置;(3)控制功能 选择好试验的种类之后,系统开始运行,按一定要求的顺序和间隔,控制机械臂、孵育仓、洗板机、读板机的开启和停止;并从整个批次试验的高度去优化动作的执行序列,让系统花最少的时间去完成整个批次的试验;(4)数据显示 图像方式实时显示试验进度;试验完成后,用表格的方式显示试验结果,并用不同的颜色表示试验结果是阴性还是阳性;(5)实时报警 试验过程中,一旦药品、清洗液、针管不够,吸液、吐液时针管堵塞或者出现其它的一些异常情况,系统都会立即给出不同的报警提示,并将报警信息记录,以备检查;(6)数据管理 可查询历史试验结果。选择日期时间后,以列表方式或表格方式显示记录;可查询报警的记录;可查询用户的登录情况以及用户对系统配置的改变所作的操作;提供了和医院病人数据库连接的接口;(7)管理功能 系统设有密码和操作人员级别管理,不同级别人员具有不同的操作权限。

1 系统的软件设计

1)软件环境要求。本系统要实现上述功能,要求实时性好,要与现场设备实时地进行信息交换,需要计算机与接口卡之间进行可靠的通信。系统采用C++ Builder 6.0进行开发,选用单文档多视图的工作框架。运用文件进行数据管理,用工具按钮进行选项操作^[2]。

2)面向对象分析。根据需求规格说明和鉴别或筛选规则,本系统拥有的对象如下:

(1)传感器对象 是系统检测的外部实体对象,它有上下限范围、设定值、实时值等属性,有采集实时数据的操作;(2)控制设备对象 包括机械臂和升降板机等,是系统控制操作的外部实体对象。有工作/停止的状态属性和吸液、吐液、取针、脱针操作命令;(3)实时数据记录对象 属性是记录的系统实时量,有对实时数据做

滤波、保存等操作;(4)控制类 根据孵育仓的温度、机械臂的坐标位置,执行的任务种类,依据控制算法,计算出对控制设备的命令类型和控制信号的大小;(5)历史数据记录集 系统运行过程中需要记忆的事件;(6)报警记录集 系统运行过程中出现并需要记忆的异常事件;(7)用户身份信息 为增加系统的安全,确定操作权限,记录用户身份信息。这是一个概念实体;(8)图像形式显示试验过程 用图像的形式来表示一个批次试验在优化了试验步骤后的执行顺序;(9)图像表示桌面布局以工作流程为背景,用图像的形式表示待检物、药品和洗涤液的量和位置;(10)报警记录查询视图 查询报警记录;(11)试验步骤及参数设置 用户操作设置的对象。用来设置吸液、吐液、取针、脱针、孵育、洗板和读板这一系列步骤的执行顺序以及参数;(12)报警器 这是个外部实体,作为一个对象。

2 系统实现

1)系统框架结构设计。针对本系统需要处理多种形式的数的特点,需要编写一个简洁、明了、不能操作出错的用户介面。系统采用VCL的多文档介面(MDI)的结构,可以同时编辑多个化验方案文件和化验实验文件,针对不同类型的多文档拥有重复相似的功能,运用面向对象思想,先建立一个拥有这些表单相同功能的基类(Base Class),其后,所有子表单类从基类继承。子表单类如果有不同的实现方法,可以覆盖基类的方法,也可建立新的方法。编程如下:

```
class TFmChildBase : public TFmBase //子窗体基类
{
    _published: // IDE - managed Components
        void _fastcall FormClose (TObject * Sender, TCloseAction & Action);
        void _fastcall FormCloseQuery (TObject * Sender, bool & CanClose);
    private: // User declarations
    protected:
        String m_filename; //该类操作的文件对象名
        bool m_IsNewFile; //是否是新建的
        virtual void SetCurrFileName(const String & Value); //方法: 设置文件对象名
    public: // User declarations
        _fastcall TFmChildBase (TComponent * Owner); //构造对象
        virtual bool Modified(void) {return false;} //返回:被操作的文件对象是否被修改
        virtual void SaveToFile(const String & fname); //保存当前文件对象到硬盘中存储
        virtual void OpenFromFile(const String & fname); //打开硬
```

盘中文件到内存中操作

```
_property String CurrFileName = {read = m-filename, write =
SetCurrFileName}; //属性:当前操作文件名
_property bool IsNewFile = {read = m-IsNewFile, write =
m-IsNewFile}; //属性:新建文件否
};
class TFmAepAssayFlow : public TFmChildBase //子表单类继承
基类时
{
Public:
    virtual bool Modified(void){return false;} //返回:被操作的
文件对象是否被修改
    virtual void SaveToFile(const String &fname); //保存当前文
件对象到硬盘中存储
    virtual void OpenFromFile(const String &fname); //打开硬
盘中文件到内存中操作
    .....
};
```

对于主窗体类,建立后设置属性:Form Style 为 fs-MDIForm,增加 TActionList 组件,建立调用 Action 列表到 TActionList 组件中,子表单创建时,就放在 Action 的 Execute 中实现。在打开多个子表单时,打开等功能只对当前活动(Active)的子表单有效。

(2)控制类的实现。控制类是实现系统在最少时间内顺利完成加样、孵育、洗板和读板这一系列试验步骤的关键环节。下面介绍试验步骤执行序列的优化算法。

问题描述:给定一个试验的集合和一个资源的集合,每种资源在同一时间最多可以执行一个试验。每个试验包括一系列步骤,每个步骤要在特定的资源上不间断地执行事先给定的时间。算法的目的就是确定一个调度,该调度将每个步骤分配到对应资源上的某个时间段,同时最小化完成所有试验所需的最小加工持续时间。

试验资源调度问题的本质是建立试验步骤在每种资源上的排列,该排列可以满足用于最小化试验执行时间的先后约束。如果能够得到这样的排列,就可以很轻松地通过依赖于问题的过程得到问题的解。将一种改进自适应遗传算法^[3]应用于文中问题。

Srinivas M. Patnaik L M(1994)提出一种自适应遗传算法(Adaptive Genetic Algorithm, AGA)^[4],能够使交叉概率 P_c 和变异概率 P_m 随群体的适应度自动改变。当种群各个体的适应度趋于一致或趋于局部最优时,使 P_c 和 P_m 增加,以跳出局部最优;而当群体适应度比较分散时,使 P_c 和 P_m 减少,以利于优良个体的生存。同时,对于适应度高于群体平均适应值的个体,选择较小的 P_c 和 P_m ,使得该优良解得以保护;而低于平

均适应值的个体,选择较大的 P_c 和 P_m 值,增加新个体产生的速度。

与一般遗传算法相比,自适应遗传算法的交叉概率与变异概率不是一个固定值,而是按群体的适应度进行自适应调整。自适应遗传算法的步骤如下^[5]:

(1) 编码/解码设计同一般遗传算法。

(2) 遵守微孔板的步骤约束条件,用随机地产生 N (N 是偶数) 个候选解等初始群体产生方法,组成初始群体。

(3) 定义适应度函数,计算各个个体的适应度 f_i 。

(4) 按照轮盘赌规则等选择方法选择 N 个个体,计算群体的平均适应度 f_{avg} 和最大适应度 f_{max} 。

(5) 将群体中的各个个体随机搭配成对,共组成 $N/2$ 对,对每一对个体,按照公式(1) 计算自适应交叉概率 p_c ,以 p_c 为交叉概率进行交叉操作,即随机产生 $R(0,1)$,如果 $R < p_c$,则对该个体进行交叉操作。

$$p_c = \begin{cases} \frac{k_1(f_{max} - f')}{f_{max} - f_{avg}} & f' \geq f_{avg} \\ k_2 & f' < f_{avg} \end{cases} \quad (1)$$

(6) 对于群体中的所有 N 个个体,按照公式(2) 计算自适应变异概率 p_m ,以 p_m 为变异概率进行变异操作,即随机产生 $R(0,1)$,如果 $R < p_m$,则对该染色体进行变异操作。

$$p_m = \begin{cases} \frac{k_3(f_{max} - f)}{f_{max} - f_{avg}} & f' \geq f_{avg} \\ k_4 & f' < f_{avg} \end{cases} \quad (2)$$

(7) 计算由交叉和变异生成的新个体的适应度,新个体与父代一起构成新一代群体。

(8) 判断是否达到预定的迭代次数,如果达到,则结束寻优过程;否则转(4)。

在式(1)、(2)中, f_{max} 是每代群体中个体的最大适应度值; f_{avg} 是每代群体中的平均适应度值; f' 是被选择交叉的两个个体中较大的适应度值; f 是被选择变异个体的适应度值。只要设定 k_1, k_2, k_3, k_4 取 $(0,1)$ 区间的值,就可以自适应调整了。

改进的自适应遗传算法是根据当代的最优个体不被破坏,仍然保留,但也要有一定的交叉概率与变异概率的思想提出的,称之为 S- 自适应算法。这种方法根据公式(3)、(4) 计算交叉概率 p_c 、变异概率 p_m 。公式(3)、(4) 中由于采用了正弦函数,所以称为 S- 自适应遗传算法。

S- 自适应遗传算法交叉概率、变异概率计算公式:

$$p_c = \begin{cases} k_1 \sin\left(\frac{\pi}{2} \frac{(f_{max} - f')}{f_{max} - f_{avg}}\right) & f' \geq f_{avg} \\ k_2 & f' < f_{avg} \end{cases} \quad (3)$$

$$p_m = \begin{cases} k_3 \frac{(f_{\max} - f)}{f_{\max} - f_{\text{avg}}} & f' \geq f_{\text{avg}} \\ k_4 & f' < f_{\text{avg}} \end{cases} \quad (4)$$

在公式(3)和(4)中,一般取 $k_1 = 1.0$, $k_2 = 1.0$, $k_3 = 0.5$, $k_4 = 0.5$ 。

3 实验结果

假设某个批次的试验里,有8块96孔微孔板,每块微孔板要经过加样→孵育→洗板→加样→孵育→洗板→加样→孵育→加样→读板10个步骤。已知每个步骤占用每个模块处理时间矩阵 T 为:

$$T = \begin{bmatrix} 6 & 30 & 3.6 & 1.5 & 20 & 3.6 & 1.5 & 15 & 1.5 & 0.5 \\ 4.8 & 30 & 3.6 & 1.5 & 30 & 3.6 & 1.5 & 15 & 1.5 & 0.5 \\ 2.8 & 40 & 1.6 & 0.9 & 30 & 1.6 & 0.9 & 15 & 0.9 & 0.23 \\ 5.2 & 30 & 3 & 1.3 & 30 & 3 & 1.3 & 15 & 1.3 & 0.41 \\ 3.6 & 40 & 3.6 & 1.5 & 30 & 3.6 & 1.5 & 15 & 1.5 & 0.5 \\ 4.4 & 30 & 2.62 & 1.1 & 30 & 2.62 & 1.1 & 15 & 1.1 & 0.38 \\ 2 & 30 & 1.2 & 0.5 & 30 & 1.2 & 0.5 & 15 & 0.5 & 1.3 \\ 4 & 30 & 2.4 & 1.0 & 30 & 2.4 & 1.0 & 15 & 1.0 & 0.27 \end{bmatrix}$$

对于上述试验任务,用改进的自适应遗传算法和随机算法分别生成试验执行序列^[6]。在使用随机算法时,该批次试验执行完毕所需时间为225.3分钟。而在使用改进的自适应遗传算法时,取 $p_m = 0.1$, $p_c = 0.6$, $N = 80$,种群规模为20,该批次试验完成的时间是132.6分钟。效率提高了69.9%。同时,在医院的试用过程当中,血样检测的准确率也达到了99.6%以上。

4 结束语

根据实际工程需要,按软件工程的设计思想,归纳

出全自动酶免工作站计算机控制系统的对象,给出系统实现单文档多视和控制类的详细设计,并将改进的自适应遗传算法应用于系统执行效率的优化。通过模拟比较实验可见,改进后的系统执行效率显著提高,实验结果对全自动酶免工作站的设计具有指导性意义。预期将会应用于新的全自动酶免工作站的设计之中。每台将自适应遗传算法应用于系统执行效率的酶免仪可以提高将近70%的执行效率,每天可以多检测120个左右的血样,如果按照现在医院血液检测的指导价,每台酶免仪每年可为医院多创造150万人民币左右的利润。

参考文献:

- [1] 金伯泉.细胞和分子免疫学实验技术[M].西安:第四军医大学出版社,2002:61-65.
- [2] Douglass B P. Real-Time Design Patterns - Robust Scalable Architecture for Real-Time System[M].北京:北京航空航天大学出版社,2004:112-119.
- [3] 王万良,吴启迪,宋毅.求解作业车间调度问题的改进自适应遗传算法[J].系统工程理论与实践,2004,20(2):58-62.
- [4] Goncalves J F, de Magalhães Mendes J J, Resende M G C. A hybrid genetic algorithm for the job shop scheduling problem[J].European Journal of Operational Research,2005,167:77-95.
- [5] Cheng R, Gen M, Tsujimura Y. A tutorial survey of job-shop scheduling problems using genetic algorithms representation[J].Computers & Industrial Engineering,1996,30(4):983-997.
- [6] 许家玉,经亚枝.基于DSP+FPGA的遗传算法硬件实现[J].微计算机信息,2005(1):127-128.

(上接第130页)

通过较小的样本量(训练样本:200个;测试样本100个)的训练与测试,识别程序已能达到94%的识别率。因限于各种因素,只对网络进行了小样本的训练,按照理论分析,随着训练样本的增加,识别率会大幅上升。可以预见,若学习1000个样本以上,识别率应能达到95%~97%。

5 结束语

提出一种基于BP神经网络的手写体数字的识别方法,通过提取手写体数字的点特征和区域笔划方向特征以及通过改进的BP算法设计神经网络分类器,实现了手写体数字的识别。实验结果证明,此方法行之有效。利用基于BP神经网络的手写体数字的识别

方法,不仅简化了传统识别的繁杂性,而且提高了识别的准确性。因此人工神经网络作为模式识别的手段,具有很大的潜力和广泛的应用前景。

参考文献:

- [1] 边肇祺.模式识别[M].北京:清华大学出版社,1999.
- [2] 阮秋琦.数字图像处理学[M].北京:电子工业出版社,2001.
- [3] 谷口庆治.数字图形处理[M].北京:科学出版社,2002.
- [4] Castleman K R. Digital Image Processing[M]. [s.l.]:Prentice-Hall International, Inc,1988.
- [5] Rumelhart D E, McClelland J L. Parallel Distributed Processing[M].Cambridge:MIT Press,1986.
- [6] 张立明.人工神经网络的模型及其应用[M].上海:复旦大学出版社,1993.