

## 第 5 章 人工代谢算法

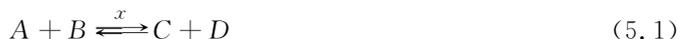
人工代谢算法是一种基于酶催化模拟生物体新陈代谢机理的仿生算法。代谢反应的核心是浓度的平衡,在酶对底物的催化效率达到最大且代谢反应实现平衡时,代谢系统的性能指标处于最优状态。如果将待优化的目标函数看作代谢反应速率,酶的催化过程则可视为对目标函数的优化过程。当反应实现平衡时,代谢速率取得稳态最大值,即目标函数取得最大值。本章简要介绍人工代谢算法的原理、编码、竞争算子、平衡算子、凋亡算子等代谢算子的描述及算法实现流程。

### 5.1 人工代谢算法的提出

人工代谢算法(Artificial Metabolic Algorithm, AMA)是 2009 年由胡杨和桂卫华以生物体新陈代谢机理为模型提出的一种基于酶催化的仿生算法。AMA 面向的对象是一个网络化的控制系统,算法通过模拟生物体新陈代谢的环境,以代谢物浓度差及底物与酶的契合程度为控制量对系统变量进行调节。它的控制目的是使整个代谢网络的流量趋于平衡和协调,这与网络控制的目的恰好一致。因此,从定性的角度来说,人工代谢算法可以实现对复杂系统、多对象多目标系统较好的实时控制。起初,人工代谢算法用于解决多对象物流配送优化问题。后来,胡杨和桂卫华又将人工代谢算法用于解决 TSP 问题、多对象调度问题及故障诊断问题。

### 5.2 人工代谢算法的原理

人工代谢系统是在分析生物体新陈代谢规律的基础上,通过对酶识别能力、酶催化下细胞各类递阶调控模式的分析和模型抽象而成的一类模拟人的代谢系统。从生物化学层次看,任何代谢系统网络都是由一些支路连接而成的。对单条代谢支路的代谢反应模式为



其中,  $A$ 、 $B$  为底物;  $C$ 、 $D$  为生成物;  $x$  为催化酶。双向箭头表示该反应可以平衡移动,是可逆反应。

若底物浓度高于生成物浓度,此时酶起到正向催化作用,反应过程向合成生成物方向进行;若生成物浓度高于底物浓度,此时酶表现为逆向催化作用,反应过程向合成底物方向进行。在此处,酶只起常规的催化作用。若遇特殊情况,即使底物浓度低于生成物浓度,也可以

通过调节酶的浓度来改变反应过程的方向,迫使反应向合成生成物方向(正向催化方向)进行,以实现整个网络的代谢平衡。对可逆反应而言,通常提到的酶是指对正反应起加速作用的催化剂。而实际上,也存在促使化学平衡向逆反应方向进行起到抑制剂的作用。为了统一起见,对酶是正、逆反应催化剂的统称,其在算法中的作用通过平衡算子来统一表征。

由于酶对底物具有较强的选择契合性,对于不同的底物,需要不同的酶来与之发生催化作用。通过设计竞争算子来体现酶与底物催化时的专一性。同时,生物体细胞发育机制研究表明,染色体端粒(Telomeres)的 DNA 序列都随着细胞分裂次数的增加而进行缩短。在真核生物染色体中具有特殊的结构端区,即端粒区。细胞每次分裂,都由一种特殊的酶即端粒酶(Telomerase)负责将全部端粒重复序列加到子染色体上。但该酶总是只在配子细胞和癌细胞中存在。在一般的体细胞中,端粒酶数量下降,端粒变得越来越短。染色体丢失的不仅是非编码的重复序列,而且还危及基因编码。染色体一旦缩短到临界长度,细胞便会死亡。这就是多细胞生物衰老的端粒学说。据此,根据酶催化底物的专一性设计了凋亡算子,随着代谢网络中特定回路平衡的实现,对应催化该回路底物的酶开始凋亡。这一过程是为了降低计算成本,避免产生不必要的代谢计算。

人工代谢算法原理的核心是浓度平衡。当各条支路及整个代谢网络实现底物与反应物浓度达到协调平衡时,新陈代谢处于最旺盛阶段,对应的目标函数也将获得最优值。其中,当目标函数达到最优值时,对应的代谢浓度将达到稳态平衡,代谢过程的反应率也将达到稳态平衡,此时的代谢物数值对应目标函数的最优值。

## 5.3 人工代谢算法的描述

### 1. 浓度平衡与代谢算子

在人工代谢系统中,底物、反应物的初始浓度由输入输出物理量决定。系统输入输出量关系决定代谢反应率的初始值。AMA 算法将“浓度平衡”即“浓度差”作为适应度函数,根据代谢支路上的代谢物的多少决定对应催化酶的浓度。

通过平衡算子和抑制算子等代谢算子对代谢浓度进行调节,浓度差越大,则调节幅度也越大。经反复循环调节,浓度的变化率为 0(即浓度差为 0),实现浓度平衡。此时系统已完全畅通,代谢量和代谢率也逐渐趋于稳定。此时从底物到生成物之间形成一条最优的代谢通路。代谢物数值达到最大,所对应的目标函数值最大而获得最优解。

凋亡算子适用于多酶调控的代谢体系,凋亡算子只取全 0 和全 1 两种情形。当酶所催化的代谢反应未达到平衡时,该酶的凋亡算子置全 0,表示酶尚未退出代谢体系;当酶所催化的代谢反应已达到平衡时,该酶的凋亡算子置全 1,表示酶退出代谢体系,在余下的反应中不再考虑该酶的作用。

### 2. 人工代谢算法的编码

设有代谢反应方程为式(5.2)、式(5.3),其中  $A$  和  $B$  为底物, $P$  和  $Q$  为生成物, $E$  为催化酶。在代谢规律中,酶  $E$  先与底物结合,生成中间复合物,再生成最终生成物。具体结合方式如式(5.3)所示。



其中,  $[AE]$  为中间复合物。酶  $E$  先与底物  $A$  结合产生中间复合物  $[AE]$ , 再释放出生成物  $P$ 、 $Q$  并还原生成催化酶  $E$ 。

基于共价催化的原理, 人工代谢算法的编码规则如下。

底物  $A$  和  $B$  的浓度区间为  $[0, 1]$ , 由底物  $A$  和  $B$  的量各占整个反应平衡方程中全部物理量的比例可计算出浓度。在人工代谢算法中, 代谢物(包括底物和生成物)和酶的编码都用二进制位串表示。

假设采用  $n$  位二进制代码的编码方式, 即将底物实际浓度和酶的浓度各映射成  $[0, 2^n - 1]$  上的 0、1 组成的编码串。中间代谢物通过底物和酶对应位置上的 0、1 值经过“同或”逻辑的操作得到。例如, 设底物  $A$  的编码为 00101101, 酶  $E$  的编码为 10110001, 则中间物  $[AE]$  的编码由  $A \odot E$  可得到, 即为 01100011。显然, 当中间物  $[AE]$  的编码达到 11111111 时, 代谢平衡予以实现。这样编码的优点在于充分体现底物与酶的契合程度, 从直观意义上进一步加强了算法自身的生化背景和寻优意义。

### 3. 竞争算子的设计

通过对生物化学中相关知识的分析, 可以发现酶的催化作用有很强的选择性。对于不同的底物有不同的选择性。设有两个不同的底物  $A_1$ 、 $A_2$ , 酶  $E$  对它们的契合程度很可能有较大的差距。通过引入竞争阈值  $h_1$  来评价契合程度, 设待优化函数  $y = f(x)$ , 其中自变量  $x$  的二进制编码(即为底物  $A$  的编码), 反过来底物  $A$  的二进制串解码成对应的十进制数(即为  $f$  函数的自变量  $x$ )。设底物  $A_1$ 、 $A_2$  分别对应的十进制数为  $x_1$ 、 $x_2$ , 当满足式(5.4)时, 说明底物  $A_1$  的代谢量大于底物  $A_2$  的代谢量, 且底物  $A_1$  与酶的契合程度要大于底物  $A_2$  与酶的契合程度, 因此底物  $A_1$  较之于  $A_2$  是更为理想的代谢物, 因此  $A_1$  应保留进入下一轮的代谢选择; 当满足式(5.5)时, 说明底物  $A_2$  的代谢量大于底物  $A_1$  的代谢量, 且底物  $A_2$  与酶的契合程度要大于底物  $A_1$  与酶的契合程度, 因此底物  $A_2$  较之于  $A_1$  是更为理想的代谢物, 因此  $A_2$  应保留进入下一轮的代谢选择。当式(5.4)、式(5.5)均不能满足时, 说明  $A_1$ 、 $A_2$  相互之间无明显的竞争优势, 因此随机选择  $A_1$ 、 $A_2$  中的任一个进入下一轮的代谢。

$$\left\{ \begin{array}{l} \left| \frac{f(x_1) - f(x_2)}{f(x_1) + f(x_2)} \right| \geq h_1 \\ \frac{f(x_1) - f(x_2)}{f(x_1) + f(x_2)} \geq 0 \end{array} \right. \quad (5.4)$$

$$\left\{ \begin{array}{l} \left| \frac{f(x_1) - f(x_2)}{f(x_1) + f(x_2)} \right| \geq h_1 \\ \frac{f(x_1) - f(x_2)}{f(x_1) + f(x_2)} \leq 0 \end{array} \right. \quad (5.5)$$

### 4. 凋亡算子的设计

设计凋亡算子的目的是淘汰与酶过分不相匹配的底物, 而对应产生新的底物。其设计过程如下。

设立凋亡阈值  $h_2$ , 对任意底物  $A_1$ , 对于已给定的酶  $E$  而言, 基于“同或”逻辑操作计算出

其中间代谢物 $[A_1E]$ ,设中间代谢物 $[A_1E]$ 对应的十进制数为 $k_1$ ,各位全为1时的中间代谢物对应的十进制数为 $k_2$ (显然,当编码位数为 $n$ 时, $k_2=2^n-1$ )。当满足式(5.6)时,说明底物 $A_1$ 与酶 $E$ 匹配程度过低,底物 $A_1$ 应予以凋亡,而用新的底物 $A_2$ 来代替。

由于代谢操作希望中间代谢物的同或结果尽可能接近全为1,因此新底物 $A_2$ 的产生过程如下。

设代谢物采用 $n$ 位编码,则底物 $A_1$ 与酶不契合的程度可定义为 $m$ 位, $m$ 值可由式(5.7)得出,其中函数 $\text{round}()$ 表示对自变量进行四舍五入取整运算。

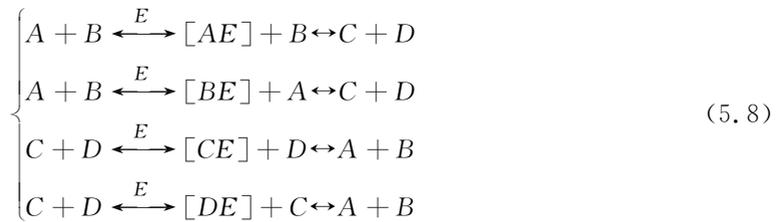
$$\frac{k_1}{k_2} \leq h_2 \quad (5.6)$$

$$m = \text{round}\left(\frac{k_1}{k_2} \times n\right) \quad (5.7)$$

得到 $m$ 后,在组成 $A_1$ 的二进制串的 $n$ 位编码中随机选择 $m$ 位编码进行取反操作,得到的结果即为 $A_2$ 。显然, $A_2$ 与 $E$ 的契合程度要高于 $A_1$ 与 $E$ 的契合程度。这时 $A_2$ 代替 $A_1$ 进入下一轮的代谢计算。当不能满足式(5.6)时,说明 $A_1$ 尚未达到凋亡的程度,则保留 $A_1$ 进入下一轮的代谢计算。

#### 5. 平衡因子和拟制因子的设计

代谢反应的平衡是维系生命活动的关键。平衡算子的目的是当代谢平衡不能持续时,产生新的代谢物来尽可能保持代谢中间物的最大化。设有底物 $A$ 、 $B$ 和生成物 $C$ 、 $D$ ,在式(5.1)中作为可逆反应,酶可完成双向催化功能。故式(5.1)等价于式(5.8)中4个反应的合成。



式中,酶是所有能催化代谢物的作用的统称。当反应实现平衡时,酶与各物质之间的总体协调功能达到最强。从式(5.8)可以看出,当酶 $E$ 确定时,4个方程中只有3个是彼此独立的,故可以由其中的任意3个代谢物推出第四个代谢物。如果设 $A$ 、 $B$ 、 $C$ 的编码已知,则中间代谢物 $[AE]$ 、 $[BE]$ 、 $[CE]$ 、 $[DE]$ 的二进制编码可以由“同或”逻辑得到。例如,由 $A$ 的二进制编码与 $E$ 的二进制编码的“同或”可以得到 $[AE]$ 的编码。

设 $P_1=[AE]$ , $P_2=[BE]$ , $P_3=[CE]$ ,当代谢实现平衡时, $P_1$ 、 $P_2$ 、 $P_3$ 应尽可能达到全为1。设 $P=P_1 \odot P_2 \odot P_3$ ,显然, $P$ 也尽可能达到全为1。再令 $P_4=[DE]$ ,由代谢平衡条件, $P$ 与 $P_4$ 应尽可能接近。故可知当达到完全理想平衡状态时有 $P_4=P$ 成立。代谢物 $D$ 的编码可用如下方法得到:逐位对比 $P$ 和 $E$ 的二进制位串,设对第 $i$ 位, $P(i)=E(i)$ 时, $D(i)=1$ ;  $P(i) \neq E(i)$ 时, $D(i)=0$ ;这样做的意义在于使 $D$ 与 $E$ 的匹配程度与 $P$ 的取值更为接近,从而实现平衡状态下新代谢物的产生和搜索。

人工代谢算法的平衡因子 $k_{m_0}$ 和拟制因子 $k_{i_0}$ 的初值分别为式(5.9)、式(5.10),平衡因子 $k_m(t)$ 和拟制因子 $k_i(t)$ 的调节规律分别为式(5.11)、式(5.12)。

$$k_{m0} = \frac{f(A+B)}{f(A+B+C+D)} \quad (5.9)$$

$$k_{i0} = \frac{f(C+D)}{f(A+B+C+D)} \quad (5.10)$$

$$k_m(t) = k_{m0} \frac{\partial f(A+B+C+D)}{\partial f(A+B)} \quad (5.11)$$

$$k_i(t) = k_{i0} \frac{\partial f(A+B+C+D)}{\partial f(C+D)} \quad (5.12)$$

$$v(t) = \frac{\partial f(A+B)}{\partial f(A+B+C+D)} + \frac{\partial f(C+D)}{\partial f(A+B+C+D)} \quad (5.13)$$

式(5.9)~式(5.12)中,  $f(\cdot)$ 表示对应物质的量,如  $f(A+B)$ 表示全部反应物的浓度之和。式(5.13)中,  $v(t)$ 为系统反应速率。式(5.9)~式(5.13)中,  $A, B, C, D$ 为不同的底物或生成物,是随时间变化的量。

在人工代谢算法中,由于在实际情况下代谢路径中的代谢物浓度不可能完全为0,因此引入数学上的某一极小值  $\epsilon$ 。当代谢物浓度小于  $\epsilon$  时,认为对应的代谢反应已实现平衡,即为

$$\frac{\partial v(t)}{\partial t} < \epsilon \quad (5.14)$$

其中,  $\epsilon$  是不为0的某一极小值。 $\epsilon$ 可以按实际优化进程进行交互式定义。

平衡算子的作用是随机选择酶  $E$  中的二进制数位,使对应的数位上的数值与底物对应数位上的数值相等,即若底物某一数位上的数值为0,则平衡算子将酶  $E$  中与底物对应数位上的数值也转化为0;若底物某一数位上的数值为1,则平衡算子将酶  $E$  中与底物对应数位上的数值也转化为1。

抑制算子的作用是随机选择酶  $E$  中的二进制数位,使对应的数位上的数值与底物对应数位上的数值不相等,即若底物某一数位上的数值为0,则抑制算子将酶  $E$  中与底物对应数位上的数值转化为1;若底物某一数位上的数值为1,则抑制算子将酶  $E$  中与底物对应数位上的数值转化为0。这种编码的思想是使酶  $E$  与底物之间的作用关系更好地与真实的“共价催化”原理相一致,从而通过调整酶  $E$  的变化来调整化学平衡。

平衡算子和抑制算子的值均在 $[0,1]$ 区间。平衡算子在一个计算周期中的值为当前全部底物的浓度;抑制算子在一个计算周期中的值为全部生成物的浓度。在得到平衡算子和抑制算子的值后,决定所要变换的酶的二进制编码的位数。当代谢反应达到稳态,代谢物浓度变化率趋于不变或小于  $\epsilon$  时,代谢反应速率已达到稳态最大值,代谢寻优终止。

## 5.4 人工代谢算法的实现流程

人工代谢算法的实现流程如图5.1所示。首先,设定基本的代谢参数,包括底物点数目  $m$ ; 酶的取值  $e$ ; 代谢计算迭代次数  $G$ ; 代谢点编码位数  $n$ ; 竞争阈值  $h_1$ ; 凋亡阈值  $h_2$ 。其次,通过对实际问题的编码,将待优化问题的自变量空间用底物的取值来表示。采用竞争、凋亡、平衡等酶的3种代谢算子计算方式来选取新的底物。根据酶与底物契合程度的高低来筛选出一批更为合适的底物进入下一轮的代谢选择。上述过程反复进行,最后通过

判断是否达到代谢计算的迭代次数或判断酶与底物契合的稳固程度来决定代谢计算是否可以完成。在完成代谢计算后,通过对反复“发酵”后得到最优底物进行解码可以得到实际问题的最优解。

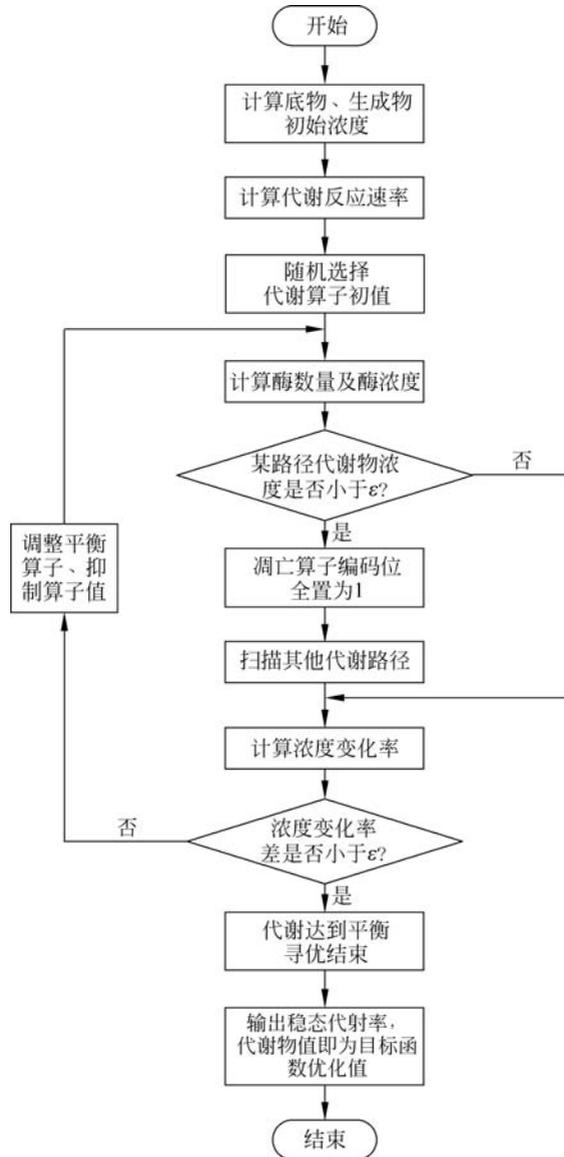


图 5.1 人工代谢算法实现流程图