**摘要**

**关键字** 约束满足问题 RB模型 模拟退火算法 遗传算法

## 引言

约束满足问题（Constraint Satisfaction Problem，CSP）是人工智能领域的研究热点，其在计算机科学、信息论等理论研究中有非常重要的地位，在实际应用领域中，例如模式识别、决策支持、物流调度及资源分配都有非常广泛的应用。

通常，一个CSP包含个变量和个约束，每个变量从其相应的定义域；每个约束包含从变量集合中随机挑选的个不同的变量，还包含一个不可兼容赋值集合，其中的元素都是上述个变量的不可兼容赋值[4]。如图着色、最小顶点覆盖以及可满足性问题（k-SAT）等都是约束满足问题。显然，可能的约束个数共有个，从这些可能的约束中随机挑选个就构成这个CSP的一个随机实例。求解这个随机实例，就是找到对这个变量的至少一组赋值，并且满足所挑选出的这个约束；或者证明此实例不存在解。一般情况下，大多数的随机CSP都是NP-完全问题。已经有许多理论证明和实验研究表明随机CSP具有可满足性相变现象，即控制参数（如约束密度）存在一个相变区域，当控制参数处于这一相变区域时，随机实例会由可满足的状态突变为不可满足状态[5~6]。经典的模拟退火算法、遗传算法以及随机游走算法等，在接近相变区域时就会变得失效，但是仍有很多学者对一些经典的启发式算法做出改进，使得算法其在接近相变区域时依旧有很不错的求解效率[7~11]。

2000年Xu等提出了一个非平凡的随机约束满足问题——RB模型。该模型是对标准CSP模型中B模型的一种改进，克服了B模型的平凡渐近无解性[17]。 通过对变量取值域和约束紧度的限制，避免了标准B模型不能产生难解实例的缺点，并且RB模型中变量的定义域会随着变量个数的增加而呈现多项式增长[12]。在文献[12]中，严格证明了RB模型不但存在可满足性相变现象，而且可以得到精确的相变点。2006年，Xu从理论上证明RB模型在相变区域生成的随机实例具有指数级的树归结复杂性，即这些实例几乎是极其难解的[13]。随后，实验上也验证了在相变阈值附近，求解难度随问题规模增加而呈指数级增长[14]。另外，因此RB模型极易产生难解实例。针对RB模型随机实例，已经有许多不错的算法被开发出来[15~16]，但是许多经典的智能启发式算法在求解RB模型随机实例时效果很差。例如，模拟退火算法和遗传算法，它们都是启发式不完备算法，其在全局寻优的问题上表现很突出。特别无约束最优化问题中，这两种算法都有很重要的应用。

本文针对经典的模拟退火算法做出了改进，提出两种新算法，将这两种新的算法应用到RB模型随机实例的求解问题上，通过数值实验，将结果与随机游走算法进行对比分析。实验结果表明，这两种新的算法对于求解RB模型随机实例都有不错的效果。

## 问题描述

设是一个含有个变量的集合，每个变量都从定义域中取值。另外，我们从中随机选出个不同的变量，用以构成约束，每一个约束都存在一个不可兼容赋值集合，设，中的元素都是元向量，其用来限定这个变量的赋值，即对于这个变量的赋值所组成的元向量如果属于，则这组赋值不满足约束；反之，则视为满足约束。通常由以下两步来构造RB模型随机实例[12]：

1. 随机可重复挑选个约束，是大于零的常数。其中，每个约束中都含有个互异的变量。
2. 对于每个约束，我们随机无重复地选取组不可兼容值，其中，，为常数。

2000年Xu等已经严格证明RB模型不仅存在可满足性相变现象，且其相变点是精确存在的[12]，即：设表示RB模型的一个随机实例是可满足的概率，则有以下结论成立：

**定理1**[12]设，如果，都是常数，并且，以及满足，那么



**定理2**[12]设，如果，都是常数，并且与满足，那么



由上述定理可知，当充分大时，RB模型的随机实例存在可满足性相变现象，即如果，RB模型的随机实例以概率1可解，如果则以概率1不可解。另外，Xu等已在文献[13~14]指出，在相变点附近RB模型随机实例的求解难度会随着变量个数的增加而呈现指数级增长，极易产生难解实例。

## 两种改进的算法

经典的模拟退火算法，是一种通用的概率算法，其思想源于固体退火过程：当固体物质温度很高时，固体内部粒子运动杂乱无序；而当温度逐渐降低时粒子又渐渐趋于有序运动。总的来说，模拟退火算法往往用来求解优化问题的最小值问题，算法过程中会不断的对变量的当前赋值进行扰动，以产生新的赋值，如果新的赋值使得目标函数变小则接受新的赋值为当前赋值，反之，则以概率接受新的赋值，其中是当前温度，为新赋值目标函数值，为当前赋值目标函数值，重复上述过程直到退火完毕或者目标函数值满足优化要求。

为了能将模拟退火算法应用到RB模型随机实例的求解问题中来，我们把问题转化为求一组赋值使得其不满足该随机实例的约束个数最少，如果找到一组赋值使得不满足约束的个数为0，则这组赋值就是此RB模型随机实例的一个解。然而经典的模拟退火算法在求解RB模型随机实例时效率很低，考虑到RB模型随机实例的特点以及模拟退火算法本身的局限性，我们认为共有以下三个原因导致其效率低下：

1. 在求解过程中并没有针对RB模型的特点来采取一些比较能明显减少目标函数值的扰动策略；
2. 模拟退火算法在求解过程中并没有有效利用已找到的最优赋值；
3. 模拟退火算法在RB模型问题求解上很依赖初始赋值。

对于第一个问题，我们把一种针对RB模型特点的扰动策略应用到了模拟退火算法中，简单来说，就是随机挑出一个当前赋值不满足的约束，再随机挑一个在此约束中出现的变量，改变当前赋值中此变量的赋值使得改变赋值后的当前赋值满足这一约束。对于第二个问题，结合模拟退火算法的特点，我们把退火思想引入到对当前赋值的扰动操作中，具体来说，在模拟退火算法求解过程的前期，温度很高时，我们希望能在大范围的解空间去寻找优良赋值那么此时采取无目的性随机扰动的策略，而在算法中后期，温度越来越低时，我们希望可以有针对性的寻找优良赋值那么此时直接将目前找到的最佳赋值作为当前赋值，再利用解决第一个问题时所采取的针对性扰动策略进行扰动。对于第三个问题，本文先运用遗传算法对多组初始的随机赋值进行优化，然后将遗传算法结束后返回的最佳赋值作为模拟退火算法的初始解。

本文针对这三个问题，在模拟退火算法的基础做出了两种改进算法，算法1是针对问题一与问题二而提出的改进算法，而算法2则是在算法1的基础上，又加入了对初始赋值的处理。

**3.1算法1**

改进的模拟退火算法步骤：

输入：二元RB模型的一个随机实例，初始温度，终止温度，在同一温度下最大迭代次数。

输出：找到的最优赋值以及其对应的最少不可满足约束的个数。

Step1：令当前温度，随机生成一组赋值，计算这组赋值不满足约束的个数作为其内能，令最优解，最少不满足约束个数。

Step2：令等于冷却表中的下一个值。

Step3：随机选择一种扰动方案，对当前赋值进行扰动。具体地说：随机一个0到1的数记为，若，则继续Step3.1，否则继续Step3.2。

Step3.1：随机取一个变量，将这个变量前面的所有变量或者后面的所有变量的值重新从定义域中随机赋予（取前面的变量还是后面的变量也是随机的）。

Step3.2：将目前为止的最优赋值作为当前赋值，随机选取一个不可满足的约束记为，随机挑选出一个中的变量，然后从可行赋值集合中随机一个赋值赋予这一变量。

Step4：计算新赋值不可满足约束的个数记作，得到

。

Step5：如果，则新赋值作为新的当前赋值；如果，则新解以概率接受其为新的当前赋值。

Step6：更新最优赋值以及最佳内能。比较当前赋值的不满足约束的个数与当前最少不满足约束的个数的大小，如果，则将最优解替换为当前赋值，最少不满足约束个数；反之，则直接继续下一步。

Step7：判断是否为RB模型的解，即判断当前最少不满足约束的个数是否为0，是则跳出程序，返回当前最优赋值即为RB模型的解；否则继续执行程序。

Step8：在温度下，重复次的扰动和接受过程，即执行Step3，Step4，Step5，Step6以及Step7。

Step9：判断温度是否到达，是则终止算法，返回当前最优解，以及最少不满足约束的个数；否则转Step2继续执行。

**3.2关于算法1的若干说明**

关于温度冷却表，本文采用的是的方式，其中是个大于0小于1的常数，是前一刻温度，为后一刻温度。

算法1是在经典模拟退火算法的基础上针对上文所诉的第一个问题与第二个问题做出的改进，经典的模拟退火算法是在全局找最优解的启发式不完备算法，算法开始时温度很高，内能很大，对变量赋值进行扰动后接受新解的概率较大。那么此时，我们希望在很大的解空间中对变量赋值进行扰动；但是当温度越来越低，我们就希望有针对性的去对变量赋值进行扰动。这就是Step3中的两个不同的扰动方案，分别对应Step3.1以及Step3.2。本文中采用了的概率函数来实现这一目的。

算法1的确针对问题一与问题二做出了有效改进，但是并没有用行之有效的方法来改善初始赋值。直接对模拟退火算法的初始赋值进行改进比较困难，所以本文利用遗传算法再在其中融入退火思想，来给出比较好的初始赋值，从而提出了算法2。

**3.3关于遗传算法与模拟退火算法结合的实现细节**

单纯遗传算法来求解RB模型随机实例时，效率也很低，事实上，遗传算法也没有针对RB模型的特点来进行求解。为此，我们把先把经典遗传算法与模拟退火算法的退火思想结合，在退火过程中，随着温度的降低的，使得程序在求解RB模型随机实例时有越来越明显的目的性，具体的是在变异操作时的采用不同变异方案来实现的。涉及到遗传算法，那么就会有编码、适应度函数、交叉操作以及变异操作的实现问题，下面就逐点说明本文在这些问题上的实现细节。

3.3.1退火思想与遗传算法的结合

经典的遗传算法进化代数是定值，加入退火思想后，总的进化代数还保持不变，但是要把进化代数依照策略分配到不同的温度下，例如：退火的温度冷却表，对应每个温度下进化的代数设为，则有总进化代数。

3.3.2编码

遗传算法的应用首先就是要把实际问题转化为可解的编码问题。本文采取的编码是把对变量的一组赋值就作为一个染色体，或者说是种群中的一个个体。

3.3.3适应度函数

遗传算法的适应度函数是彰显个体竞争优势的，所以它的值越大越好。本文这里采用的是不满足约束个数的倒数作为适应度函数，即：设是一组赋值，是适应度，是这组赋值不满足的约束的个数，则有：。

3.3.4选择操作

选择操作是遗传算法中体现优胜劣汰思想的地方，本文采取的是轮盘赌的方法进行选择操作，所谓轮盘赌，就是个体适应度在种群中的占比作为该个体被选到的概率，即个体被选到的概率是：。

3.3.5交叉操作

交叉是遗传算法的灵魂所在，这里采取随机交叉法：随机选出两个个体，随机取出若干变量，然后将这两个个体在这些变量位置进行交叉。

3.3.6变异操作

变异是维持种群多样性的策略。本文采取了两种变异方案。方案一：随机选择基因个数与位置，再从这些变量的定义域中随机选择变量赋值实现变异。方案二：将找到的最优个体作为当前赋值，随机找当前赋值不满足的一个约束，再随机挑一个在此约束中出现的变量，改变当前赋值中此变量的赋值使得改变赋值后的当前赋值满足这一约束。这两种变异方案，并非是等概率实现的，采用的退火思想就集中体现在这里，当温度很高的时候，我们希望染色体能够跳出局部最优解，那么也就是方案一发生的概率大一些，直到温度很低时，我们要有目的性的去减少最佳染色体的不满足约束的个数，这样方案二的概率就大一些。本文采用的是3/T作为方案二发生概率，则显然温度越低，方案二发生的概率越大。

3.3.7关于参数的选择

首先因为涉及到两个退火过程，所以在算法一开始要传入两个温度控制变量以及，这两个参数分别对应着两次退火过程，也即：利用这两个参数可以产生两个温度冷却表，方法同3.2中的说明。另外，遗传算法涉及的参数较多，进化代数、种群规模、交叉概率以及变异概率等，结合RB模型的特点，本文所取的进化代数并非一直不变的，我们希望在遗传算法的后期比前期进化代数多一些，所以在温度下的进化代数计算公式是：



其中是初始温度时进化代数，是进化代数控制参数，本文中采用的是，。种群规模、交叉概率以及变异概率等参数，都可以按照以往经验来取，然后通过实验效果不断调整，直到参数合适为止。

3.3.8优化初始赋值后的模拟退火算法

在得到遗传算法优化后的初始赋值后，本文并不是直接利用算法1中所提到改进的模拟退火算法继续求解的。在算法1中，Step3.1给出的扰动策略，是为了能在较大的解空间去寻找较好赋值，但是在这里，我们已经有了遗传算法优化后的初始赋值，所以将这一扰动策略改为遗传算法变异操作中方案二的策略，即：随机挑出一个当前赋值不满足的约束，再随机挑一个在此约束中出现的变量，改变当前赋值中此变量的赋值使得改变赋值后的当前赋值满足这一约束。

**3.4算法2**

遗传算法与模拟退火算法结合算法步骤：

输入：二元RB模型的一个随机实例以及初始进化代数，种群规模popsize，交叉概率pcross，变异概率pmutation，初始温度,终止温度，进化代数控制参数，温度控制参数。

输出：一组满足所有约束的赋值或者返回不可满足

Step1：令当前温度，进化代数，当前进化代数，随机初始化种群；

Step2：判断当前温度是否大于终止温度，是则跳至Step6，否则继续下一步；

Step3：种群进化；

Step3.1：计算种群中个体的适应度，更新最优个体以及最优适应度；

Step3.2：轮盘赌法选择优势个体：计算每个个体在种群中适应度比重，随机可重复以各个个体在种群中适应度比重抽取popsize个新个体，作为新种群。

Step3.3：对上步中得到的新种群进行交叉操作：随机popsize个0到1的实数，统计小于交叉概率pcross的实数个数记为nums\_cross，从种群中随机选择nums\_cross对个体进行交叉。对于每对个体，都从1到中随机一个整数nums，然后随机无重复的选出nums个变量后，对应交换这两个个体上对应变量的赋值，然后替换掉交叉前的这对个体。将这nums\_cross对个体按照上述做法进行交叉操作后得到新种群。

Step3.4：对上步中得到的新种群进行变异操作：随机popsize个0到1的实数，这些实数中小于pmutation的实数对应的种群个体都进行Step3.4.1~Step3.4.3的变异操作。

Step3.4.1：随机一个0到1的实数记为，如果，则进行Step3.4.2，否则进行Step3.4.3。

Step3.4.2：随机选择个体中的要发生变异的变量个数与变量，再从这些变量的定义域中随机选择变量赋值，替换原个体。

Step3.4.3：将找到的最优个体覆盖为当前个体，随机找当前个体不满足的一个约束，再随机挑一个在此约束中出现的变量，改变当前个体中此变量的赋值使得改变赋值后的当前个体满足这一约束，替换原个体。

Step4：判断最优适应度是否为0，是则终止程序输出当前最优个体，即为满足所有约束的赋值，否则继续下一步。

Step5：当前进化代数，判断是否，是则继续Step3，否则令当前温度等于温度冷却表中的下一温度，令进化代数，令后继续Step2。

Step6：判断最优适应度是否为0，是则终止程序输出当前最优个体，即为满足所有约束的赋值，否则继续下一步，即开始改进的模拟退火算法。

Step7：令当前温度，将Step6中产生的最优个体作为当前赋值，计算这组赋值不满足约束的个数作为其内能，令最优赋值，最少不可满足约束个数。

Step8：令等于冷却表中的下一个值。

Step9：随机选择一种扰动方案，对当前赋值进行扰动。具体地说：随机一个0到1的数记为，若，是则继续，否则将目前为止的最优赋值作为当前赋值后继续。对当前赋值随机选取一个不可满足的约束记为，随机挑选出一个中的变量，然后从可行赋值集合中随机一个赋值赋予这一变量。

Step10：计算新赋值不可满足约束的个数记作，得到

。

Step11：如果，则新赋值作为新的当前赋值；如果，则新解以概率接受其为新的当前赋值。

Step12：更新最优赋值以及最佳内能。比较当前赋值的不满足约束的个数与当前最少不满足约束的个数的大小，如果，则将最优解替换为当前赋值，最少不满足约束个数；反之，则直接继续下一步。

Step13：判断是否为RB模型实例的解，即判断当前最少不满足约束的个数是否为0，是，则跳出程序，返回当前最优赋值即为RB模型实例的解；否，则继续执行程序。

Step14：在温度下，重复次的扰动和接受过程，即执行Step9，Step10，Step11，Step12以及Step13。

Step15：判断温度是否到达，是则终止算法，返回当前最优赋值，以及最少不满足约束的个数；否则转Step8继续执行。

* 1. **随机游走算法**

本文给出随机游走算法作为算法1与算法2的对比算法。

随机游走算法步骤：

输入：二元RB模型的一个随机实例以及最大循环次数

输出：一组满足所有约束的赋值或者，返回不可满足

Step1：当前次数时：随机对所有变量赋值；

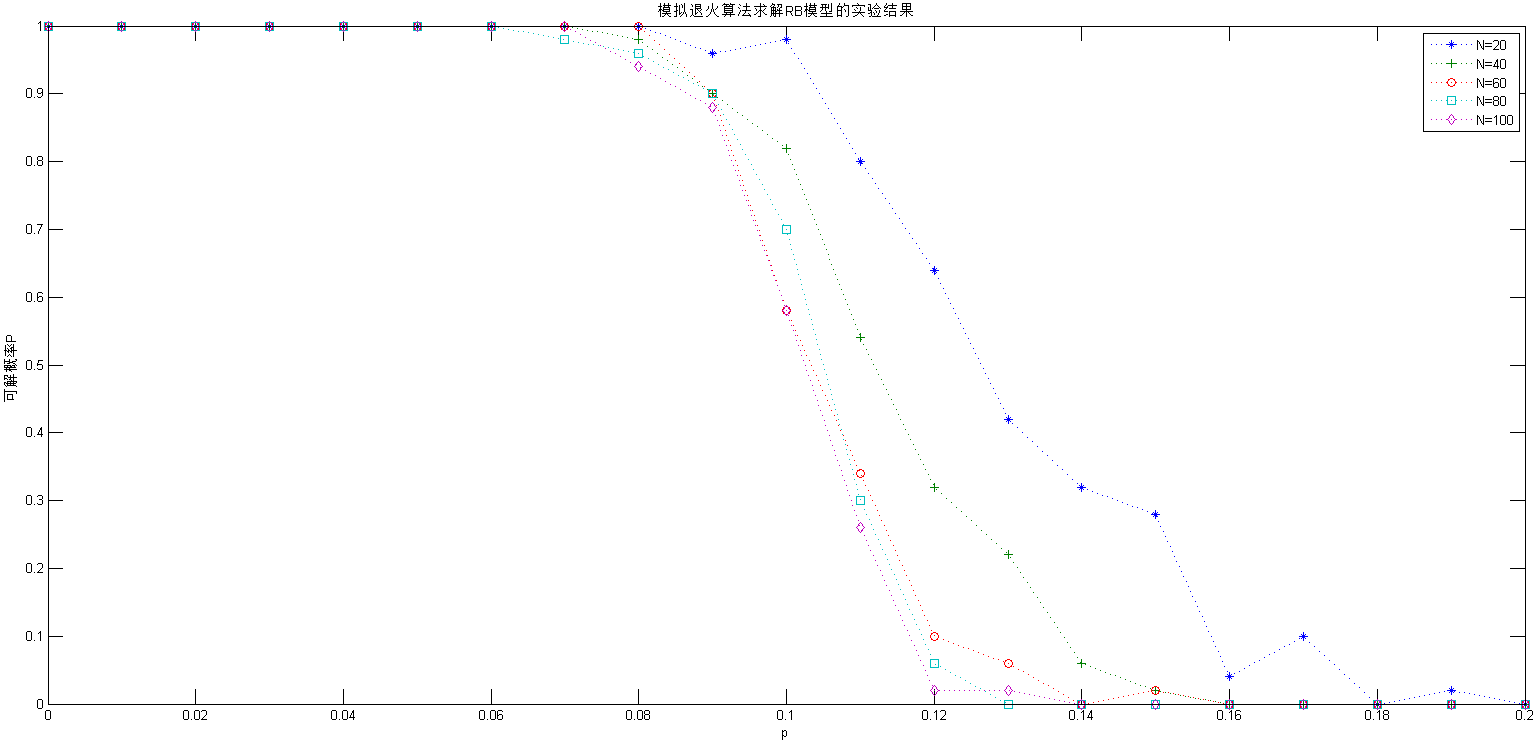
Step2：随机选取一个不满足的约束记为；

Step3：随机挑选出一个中的变量，然后从可行赋值集合中随机一个赋值赋予这一变量；Step4：判断是否满足所有约束，是则跳出循环，返回对所有变量的赋值；否则令，执行Step5；

Step5：判断是否，是则重复执行Step2，Step3，Step4；否则，返回不可满足。

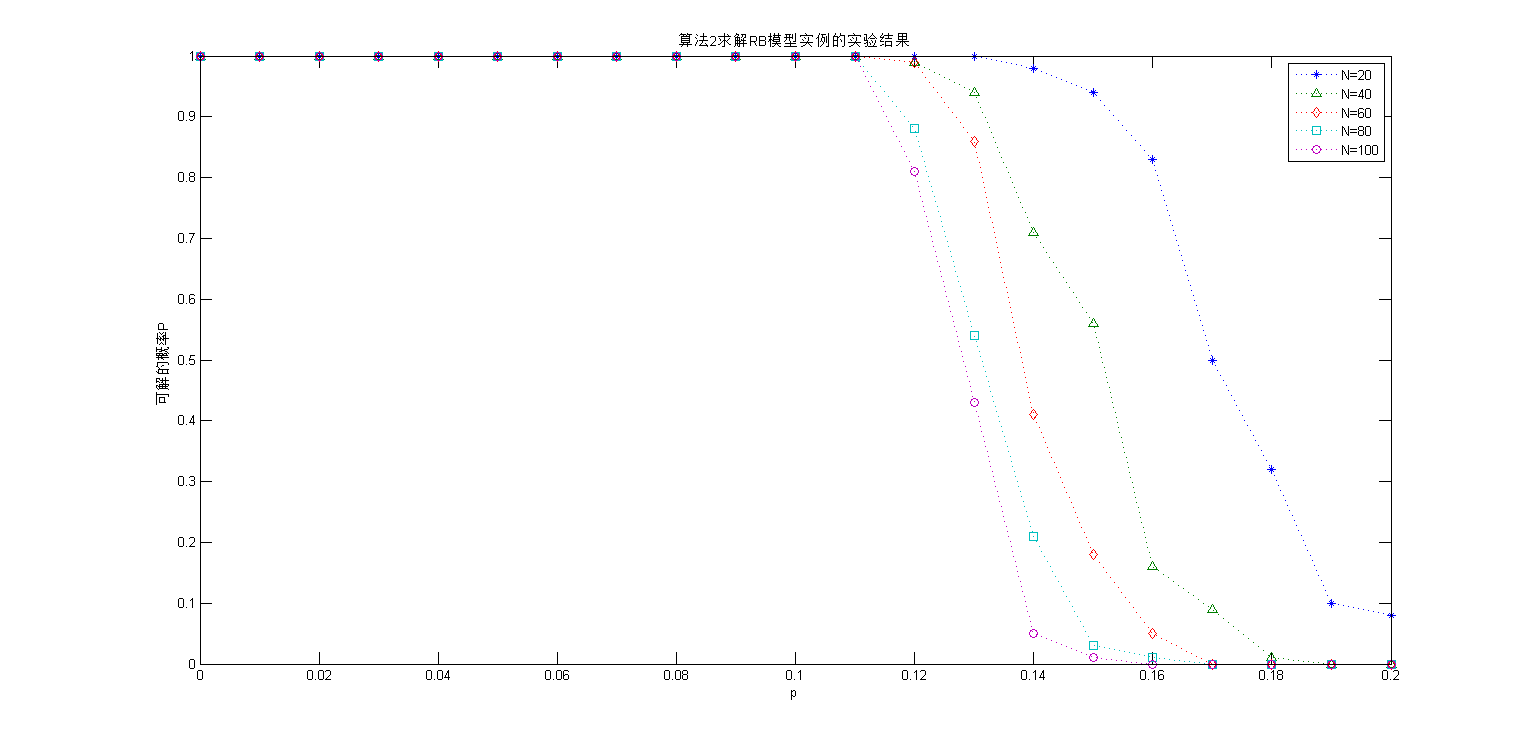
## 数值结果与算法分析

**4.1算法1的实验结果：**



上图是采用参量，取，N分别取20,40,60,80,100的参数各生成50组RB模型实例，再采用为参量的算法1求解得来的可满足性概率图。

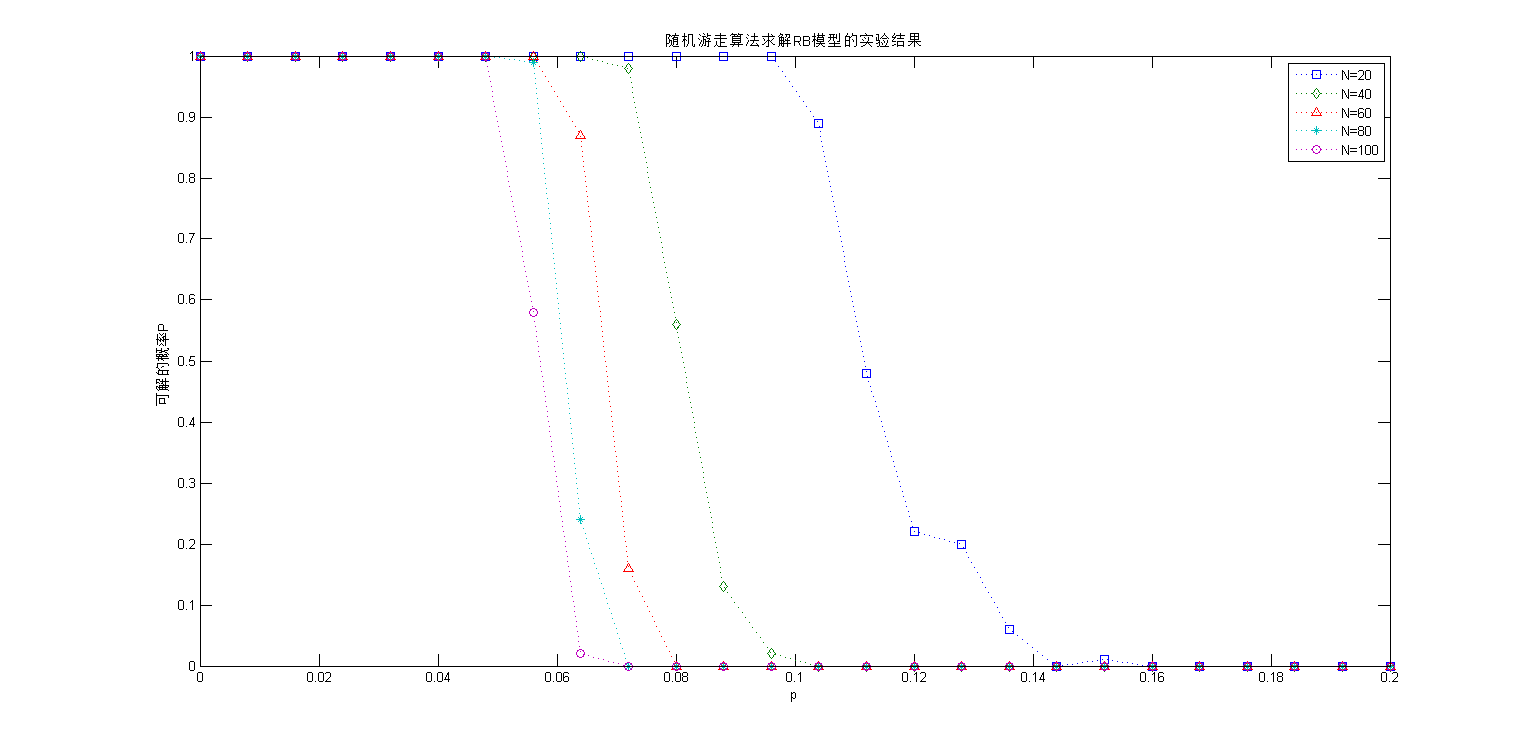
**4.2算法2的实验结果：**



上图是采用参量，取，N分别取20,40,60,80的参数各产生100组RB模型随机实例，采用的算法2的其他参数设定：初始进化代数，进化代数控制参数，种群规模popsize=50，交叉概率pcross=0.8，变异概率pmutation=0.2，初始温度,终止温度，温度控制参数，。

**4.3算法3的实验结果：**

随机游走算法的实验结果：



上图是采用参量，取P=0:0.08:0.2，N分别取20,40,60,80,100的参数生成100组RB模型实例求解得来的可满足性概率图。

**4.4算法分析**

## 结论

本文深入研究了模拟退火算法与遗传算法，针对一种具有增长定义域的随机约束满足问题，做出了两种改进算法。算法1是将退火思想的温度控制加入到变量赋值的扰动操作中，使得算法前期对解空间进行较为全面的搜索，而在算法后期进行更有目的性的搜索。算法2是在算法1的基础上，利用遗传算法优化初始赋值，然后再利用改进的模拟退火算法求解。将这两种算法运用到求解RB模型的随机实例中来，从实验结果可以看出，算法1具有不错的求解效果，而算法2的求解效果更佳突出。我们希望新算法可以为研究启发式不完备算法提供一些参考，从而促进求解约束满足问题的算法的研究。

## 参考文献

[1] Martin O C, Monasson R, Zecchina R. Statistical mechanics methods and phase transitions in optimization problems .TheorComputSci, 2001, 265: 3–67

[2]Creignou N, Daud´e H. The SAT-UNSAT transition for random constraint satisfaction problems. Discrete Math, 2009,309: 2085–2099

[3]Achlioptas D, Moore C. Random k-SAT: two moments suffice to cross a sharp threshold. SIAM J Comput, 2006, 36:740–762

[4]Tsang E. A glimpse of constraint satisfaction. ArtifIntell Rev, 1999, 13: 215–227

[5] Xu K, Li W. The SAT phase transition. Sci China Ser E: Tech Sci, 1999, 42: 494–501

[6] Fan Y, Shen J. On the phase transitions of random k-constraint satisfaction problems. ArtifIntell, 2011, 175: 914–927

[7]Huang Wen-Qi, Jin Ren-Chao . The quasi-physical person ification algorithm for solving SAT problems Solar . Science in China, Series E, 1997, 2: 179- 186 ( in Chinese)

[8]E.C.Freuder. A sufficient condition for backtrack-free search. Journal of the ACM , 1982 , 29(1) : 24-32

[9]R.J.Wallace and D.Grimes.Experimetal studies of variable selection strategies based on constraint weights. Journal of Algorithms,2008,63(1-3):114-129

[10]徐云.SAT问题的随机算法及其相变现象研究.中国科技大学博士学位论文,2002年4月

[11]杨青,马军,遗传算法用于NP完全问题的求解,山东大学学报(自然科学版),2001,6:36(2)

[12]Xu K, Li W. Exact phase transitions in random constraint satisfaction problems. J ArtifIntell Res, 2000, 12: 93–103

[13] Xu K, Li W. Many hard examples in exact phase transitions. TheorComputSci, 2006, 355: 291–302

[14]Xu K, Boussemart F, Hemery F, et al. Random constraint satisfaction: easy generation of hard (satisfiable) instances.ArtifIntell, 2007, 171: 514–534

[15]Zhao C, Zhou H, Zheng Z, et al. A message-passing approach to random constraint satisfaction problems with growingdomains. J Stat Mech, 2011: P02019

[16]赵春艳, 郑志明. 一种基于变量熵求解约束满足问题的置信传播算法. 中国科学: 信息科学, 2012, 42: 1170-1180, doi:10.1360/112011-693

[17]Achlioptas D, Molloy M S O, Kirousis L M, et al. Random constraint satisfaction: a more accurate picture. Constraints,2001, 6: 329–344