

具有同时送货和取货需求的车辆路径问题的蚁群禁忌混合优化算法*

Hybrid Optimal Algorithm of Ant Colony and Tabu Search for Vehicle Routing Problem with Simultaneous Delivery and Pick-up

殷佳林, 蒋 泰

YIN Jia-lin, JIANG Tai

(桂林电子科技大学计算机与控制学院, 广西桂林 541004)

(School of Computer Science and Control, Guilin University of Electronic Technology, Guilin, Guangxi, 541004, China)

摘要: 在分析具有同时送货和取货需求的车辆路径问题(VRPSDP)的基础上, 建立 VRPSDP 数学模型, 提出一种新的求解 VRPSDP 问题的蚁群禁忌混合优化算法(ACO-TS), 并通过实验验证该算法的有效性和可行性。新算法首先采用蚁群算法产生阶段最优解, 然后利用禁忌搜索算法对阶段最优解进一步优化。实验表明, 新算法能够高效解决 VRPSDP 问题, 并且具有较好的优化效果。

关键词: 车辆路径问题 VRPSDP 蚁群算法 禁忌搜索

中图分类号: TP301.6 **文献标识码:** A **文章编号:** 1002-7378(2008)04-0279-05

Abstract: Based on the analysis of VRPSDP, a mathematical model is built. A combined optimization algorithm of ant colony and tabu search for solving VRPSDP is designed. First, ant colony algorithm is adopted to produce a stage solution; Second, the tabu search is used to optimize the stage optimal solution. The results show that the algorithm can solve VRPSDP efficiently and obtain preferable solution.

Key words: vehicle routing problem, VRPSDP, ant colony algorithm, tabu search

近年来, 具有同时送货和取货需求的车辆路径问题(VRPSDP)备受国内外研究学者的关注, 它是对经典车辆路径问题(VRP)的一个重要扩展。VRP 解决了只送货或只取货的车辆路径问题, 然而实际物流配送中还存在着具有同时送货和取货需求的车辆路径问题。VRPSDP 既考虑了客户需要的货物从配送中心送到各个客户, 又考虑将客户供应的货物从客户处取回配送中心, 并且要求车辆能够取货和送货同时进行, 客户不需要被访问两次。VRPSDP 目前已经被广泛应用于快递物流、邮件配送、啤酒和饮料行业物流配送等行业。

VRPSDP 类问题已被证明是 NP-Hard 难题, 目前针对 VRPSDP 研究的文献较少。1989 年由 Min 在文献[1]中提出 VRPSDP, 他采用先对顾客聚类, 然后解决每一类顾客群体的 TSP 问题, 在此基础上对不可行路径进行补偿, 从而优化各 TSP 路径; 文献[2]设计求解 VRPSDP 的禁忌搜索算法, 采用四种不同类型的邻域操作以及两种不同的搜索策略; 文献[3]用并行插入算法来快速地获得 VRPSDP 问题的初始解, 然后基于模拟退火算法、禁忌表以及路线改进策略构造一种混合启发式算法; 文献[4]详细地描述 VRPSDP 与 VRPB、VRPMDP 的关系, 并基于禁忌搜索算法与导引式搜索算法设计求解 VRPSDP 的混合优化算法。

蚁群算法(ACO)是一种新的仿生类进化算法。意大利学者 Dorigo 等^[5]于 20 世纪 90 年代初提出基于蚁群种群的优化算法, 该算法通过对现实生活中

收稿日期: 2008-10-20

作者简介: 殷佳林(1982-), 女, 硕士研究生, 主要从事物流配送智能优化研究。

* 国家电子信息产业发展基金项目(信部运[2006]634号)资助。

的蚂蚁觅食行为的研究,发现蚂蚁间借助信息素进行信息的交流和传递,通过正反馈、分布式协作来寻找最优路径。禁忌搜索(TS)的思想最早由 Glover 于 1986 年提出,它是对局部领域搜索的一种扩展,是一种全局逐步寻优的启发式搜索方法算法。本文给出 VRPSDP 的数学模型,并将 ACO 算法和 TS 算法组合起来,提出求解 VRPSDP 问题的一种新的蚁群禁忌混合优化算法(ACO-TS),并进行实验分析,验证 ACO-TS 算法的有效性和可行性。

1 VRPSDP 的数学模型

1.1 问题描述

VRPSDP 问题可以描述为:配送中心用多辆车辆将客户需要的货物从配送中心送到各个客户,并将客户供应的货物从各个客户取到配送中心,要求合理安排车辆配送路线,使目标函数得到优化,并满足如下约束条件:(1)配送中心和每个客户的位置坐标是确定的;(2)每个客户取、送货量为确定值,而且取货量和送货量分别小于车辆载重量;(3)配送中心采用同一种车辆,其载重量一定;(4)在车辆路径的每条弧上,货物装载量不超过车辆的额定载重量;(5)每个客户的需求必须满足只能被访问一次且只能由一辆车服务;(6)每条配送路径的长度不超过配送车辆一次配送的最大行驶距离;(7)每辆车的起点和终点都是配送中心;(8)费用矩阵是对称的。

1.2 数学模型

VRPSDP 数学模型的目标函数为

$$\min \sum_{k=1}^K \sum_{i=1}^N \sum_{j=1}^N C_{ijk} d_{ij} X_{i \rightarrow j}^k$$

约束条件为

$$(1) \sum_{i=0}^N \sum_{k=1}^K X_{i \rightarrow j}^k = 1, \text{ 其中 } j=1, 2, \dots, N;$$

$$(2) \sum_{i=0}^N \sum_{k=1}^K d_{ij} X_{i \rightarrow j}^k \leq MD, \text{ 其中 } k=1, 2, \dots, K;$$

$$(3) \sum_{i=0}^N X_{i \rightarrow j}^k - \sum_{i=0}^N X_{j \rightarrow i}^k = 0, \text{ 其中 } j=1, 2, \dots, N, k=1, 2, \dots, K;$$

$$(4) \sum_{j=1}^N X_{0 \rightarrow j}^k \leq 1, \text{ 其中 } k=1, 2, \dots, K;$$

$$(5) \sum_{i=0}^N \sum_{k=1}^K X_{j \rightarrow i}^k P_{j \rightarrow i}^k - \sum_{i=0}^N \sum_{k=1}^K X_{i \rightarrow j}^k P_{i \rightarrow j}^k = p_j, \text{ 其中 } j=1, 2, \dots, N;$$

$$(6) \sum_{i=0}^N \sum_{k=1}^K X_{i \rightarrow j}^k D_{i \rightarrow j}^k - \sum_{i=0}^N \sum_{k=1}^K X_{j \rightarrow i}^k D_{j \rightarrow i}^k = d_j, \text{ 其中 } j=1, 2, \dots, N;$$

$$(7) \sum_{k=1}^K X_{i \rightarrow j}^k (P_{i \rightarrow j}^k + D_{i \rightarrow j}^k) \leq Q, \text{ 其中 } i \neq j, i, j=1, 2, \dots, N;$$

$$(8) \sum_{i=1}^N X_{0 \rightarrow i}^k D_{0 \rightarrow i}^k = \sum_{i=0}^N \sum_{j=1}^N X_{i \rightarrow j}^k d_j, \text{ 其中 } k=1, 2, \dots, N;$$

$$(9) \sum_{i=1}^N X_{i \rightarrow 0}^k P_{i \rightarrow 0}^k = \sum_{i=0}^N \sum_{j=1}^N X_{i \rightarrow j}^k p_j, \text{ 其中 } k=1, 2, \dots, N;$$

$$(10) d_{0j} \leq \frac{MD}{2}, \text{ 其中 } j=1, 2, \dots, N;$$

$$(11) 0 \leq \sum_{k=1}^K X_{i \rightarrow j}^k P_{i \rightarrow j}^k \leq Q, 0 \leq \sum_{k=1}^K X_{i \rightarrow j}^k D_{i \rightarrow j}^k \leq Q,$$

其中 $j=1, 2, \dots, N;$

$$(12) X_{i \rightarrow j}^k \in \{0, 1\}, \text{ 其中 } i, j=1, 2, \dots, N.$$

VRPSDP 数学模型中参数的含义为: N 表示客户点的总数; R 表示客户点的集合,各客户点编号分别为 $1, 2, \dots, N$; d_{ij} 表示节点 i 和节点 j 的距离; C_{ijk} 表示第 k 辆车从节点 i 驶向节点 j 时单位距离的运输成本; d_i 表示配送车辆在第 i 个客户点处的送货量; p_i 表示配送车辆在第 i 个客户点处的取货量; Q 表示配送车辆的最大载货能力; K 表示配送中心中配送车辆的总数; V 表示配送车辆的集合,各车辆的编号分别为 $1, 2, \dots, K$; MD 表示配送车辆的最大行驶距离; $P_{i \rightarrow j}^k$ 表示第 k 辆车从节点 i 驶向节点 j 时,该车承载的已取货物的总量; $D_{i \rightarrow j}^k$ 表示第 k 辆车从节点 i 驶向节点 j 时,该车承载的未送货物的总量; $X_{i \rightarrow j}^k$ 表示第 k 辆车是否从节点 i 直接驶向节点 j ,如果是其值为 1,否则为 0。

约束条件说明:(1)确保每个客户点仅由一辆配送车辆提供服务;(2)保证每辆配送车辆的行驶距离不能超过其最大的行驶距离;(3)保证到达每个客户点的配送车辆必然从该客户点驶出;(4)确保每辆配送车辆最多只能使用一次;(5)表示车辆从节点 i 直接到节点 j 时车辆取货物量的变化;(6)表示车辆从节点 i 直接到节点 j 时车辆送货物量的变化;(7)确保任何一条配送车辆的行驶路径上的载货量都必须小于或等于车辆的最大装载能力;(8)保证每辆车离开配送中心时的车辆的装载量为各个节点送货物量之和;(9)保证每辆车回到配送中心时的车辆装载量为各个节点取货物量之和;(10)满足最远客户的需求;(11)确保配送车辆在任何行驶路径上承载的已取货物的总量或未送货物的总量都满足车辆最大装载能力的限制。

2 求解 VRPSDP 的蚁群禁忌混合优化算法

2.1 蚁群算法与禁忌搜索的混合策略

ACO 的正反馈机制使其具有强大的局部搜索性能,但是其算法存在两个不足:第一搜索时间较长;第二易于出现早熟停滞现象使算法陷于局部最优。因此,在求解 VRPSDP 时,我们将 TS 算法引入到 ACO 算法中。TS 算法通过引用一个灵活的存储结构和相应的禁忌准则来避免迂回搜索,在搜索过程中利用禁忌表产生的记忆过程来寻找新的领域,以扩大搜索空间,并通过藐视准则来赦免一些被禁忌的优良状态,进而保证多样化的有效探索以最终实现全局优化。

ACO-TS 算法首先采用 ACO 算法产生阶段最优解,然后利用 TS 算法的关键技术对阶段最优解进一步优化,用 TS 算法强大的局部搜索能力指引 ACO 搜索更有效、更广阔的解空间,帮助 ACO 跳出局部最优,避免早熟停滞现象的发生,以此来增强 ACO 的整体寻优能力,从而提高算法的搜索性能和效率。

2.2 求解 VRPSDP 的蚁群禁忌混合优化算法

求解 VRPSDP 的 ACO-TS 算法的关键技术包括初始化信息,路径转移规则,轨迹更新规则,负反馈机制的设计以及 TS 算法中的禁忌对象及禁忌长度、邻域操作、候选集合、特赦规则、终止准则。

2.2.1 初始化信息素

采用最近邻居法(NNH)构造初始解 s_0 :相继产生一个路径的集合,每个路径开始于没有路径的、距离配送中心最近的客户,然后距离这个客户最近的可接纳的客户被加入,加入的条件以不违背车辆的装载能力和最大行驶距离为约束。当没有客户可以加入时,按照上述方法构造新的路径,或者客户都已经加入,程序停止。然后初始化信息素 τ_0 ,

$$\tau_0 = 1/n \times f(s_0), \quad (1)$$

其中 n 为客户数量。

2.2.2 路径转移规则

借鉴 Dorigo 等的 Ant-Q 算法思想^[5],将确定性与随机选择结合起来选择策略。

$$j = \begin{cases} \arg \max_{k \in \pi} \{ [\tau_{ij}]^{\alpha} [\eta_{ij}]^{\beta} [u_{ij}]^{\gamma} [\varphi_j]^{\delta} \}, q \leq r_0; \\ J, \text{ 否则;} \\ J: P_{ij}^{\delta} = \end{cases} \quad (2)$$

$$\begin{cases} \frac{[\tau_{ij}]^{\alpha} [\eta_{ij}]^{\beta} [u_{ij}]^{\gamma} [\varphi_j]^{\delta}}{\sum_{k \in \pi} [\tau_{ik}]^{\alpha} [\eta_{ik}]^{\beta} [u_{ik}]^{\gamma} [\varphi_k]^{\delta}}, \text{ 如果 } i \in \pi; \\ 0, \text{ 否则;} \end{cases} \quad (3)$$

其中 r_0 是一个常数 ($0 \leq r_0 \leq 1$), r_0 越小,蚂蚁进行随机选择的概率就越高。 q 为在 $[0, 1]$ 上均匀分布的随机数。 $\pi = \{i \in V_1 : V_1 \text{ 是可选节点} \} \cup \{0\}$ 表示节点 i 的候选点集合,其中节点 0 代表配送中心。 τ_{ij} 是节点 i 和节点 j 之间的信息素量。 $\eta_{ij} = \frac{1}{d_{ij}}$ 是节点 i 和节点 j 之间的距离的倒数。 $u_{ij} = d_{0j} + d_{0i} - d_{ij}$ 为考虑任务点与配送中心间的距离而引入,它反映两点直接连接与将两点分别与配送中心连接所能获得的路径长度的节约量。 $\varphi_j = \frac{Q_j + p_j - d_j}{Q}$ 是考虑车辆容量约束而引入的变量,它越大表示车辆利用率越高,因此选择节点 j 的概率就应该越高。当 $q \leq r_0$ 时,按照(2)式选择最好的弧,否则按照(3)式选择。

2.2.3 轨迹更新规则

轨迹更新规则包括局部更新和全局更新。局部更新是指蚂蚁在构造解的过程中,每移动一步都要对相应的弧段上的信息素进行局部更新,这样一方面可以在一个迭代周期内加强各蚂蚁之间的协作;另一方面若只有全局更新而没有局部更新,则本次迭代周期内所有的蚂蚁将在前一次全局更新过的最好路径的有限相邻区域内搜索。因此局部更新可以用来加强对未被访问弧的搜索,在一定程度上防止算法停滞。在构造路径时,蚂蚁对其访问过的每条弧采用公式(4)中给出的局部信息素更新规则来改变信息素值,

$$\tau(i, j) = (1 - \rho) \cdot \tau(i, j) + \rho \cdot \tau_0, \quad (4)$$

(4)式中 ρ ($0 \leq \rho \leq 1$) 是信息挥发参数。

全局信息素更新的目的是在最短路线上加入更多的信息素,这是一个正反馈的过程,它是应用全局信息素更新规则 $\tau(i, j) = (1 - \alpha) \cdot \tau(i, j) + \alpha \cdot \Delta\tau(i, j)$ 来改变最短路线上的所有的弧 (i, j) 相关联的信息素值,其中 α ($0 \leq \alpha \leq 1$) 是信息素挥发参数。

$$\Delta\tau(i, j) =$$

$$\begin{cases} \frac{1}{L_{gb}}, \text{ 如果弧 } (i, j) \text{ 属于全局最短路线;} \\ 0, \text{ 否则。} \end{cases} \quad (5)$$

(5)式中 L_{gb} 是目前得到的全局最优解的路线长度。

2.2.4 负反馈机制的设计

ACO 的选择机制就是使好的路径会以较大的概率选中。而正反馈机制的存在又必然会促使这些路径在以后的选择中更具竞争优势。当搜索陷入局

部最优解时,按传统的方式算法无法从局部极小值点跳出来,为此引入负反馈机制。方法为:(a)借鉴MIN-MAX的思想^[6]将各个路径上的信息量限定在某个固定的范围内,即 $MIN \leq \tau(i, j) \leq MAX$ 。这样可以减少路径之间信息度差距,促使算法从局部极小值点跳出来。(b)进行信息素全局更新时,若发现陷入局部极小值点,则加入负反馈信息量,以减少局部解对应的信息素量。

2.2.5 TS算法关键技术

TS算法的使用是基于ACO算法的,因此评价函数和ACO算法相同:以目标函数作为评价函数,目标函数越小,则解越优。在解的邻域搜索中要保证解的可行性,即满足最大行驶距离限制和最大装载量限制这两个约束条件。

2.2.5.1 禁忌对象及禁忌长度 禁忌对象是指禁忌表中的局部最优解。将每次迭代得到的最好解作为禁忌对象放入禁忌表中,并修改其任期,刷新禁忌表。禁忌长度指被禁忌对象不允许被选取的迭代步数。我们把某一解视为禁忌对象。固定禁忌长度在求解过程中保持不变。

2.2.5.2 邻域操作 采用1-0节点转移法来进行路线间的改善,该方法将节点从原路径中除去,在考虑车容量及最大行驶距离限制下,将该节点加入另一路径内;而路线内的改善是2-opt节线交换法进行,该方法是从原路线中删掉两条弧,用两条新弧代替旧弧。

2.2.5.3 候选集合 当前解的邻域中经过1-0节点转移法和2-opt节线交换法得到的满足约束条件的集合再加上解禁忌解所组成的集合。

2.2.5.4 特赦规则 采用基于评价值的特赦规则,即若候选集合中所有解都为禁忌解时,则解禁其中的最好解。

2.2.5.5 终止准则 采用ACO算法求解出来的最优解作为初始解,所以TS算法的初始解已经在最优解附近。因此迭代次数取得稍小,而候选集则取得稍大。

2.2.6 求解VRPSDP的ACO-TS算法执行流程

步骤1 设置ACO算法最大迭代次数 I^{max} 、当前迭代次数 $I=0$ 和蚂蚁数 m ,TS算法最大迭代次数 i^{max} 、当前迭代次数 $i=0$ 、参数信息素挥发系数 ρ 、阈值 r_0 、随机数 q 以及相对影响因子 $\alpha, \beta, \gamma, \lambda$ 。

步骤2 初始化信息素 τ_0 ,按公式(1)设置每条弧上信息素 $\tau(ij)=\tau_0$,设置终止条件。

步骤3 构造路线:对于蚂蚁访问的每个客户

节点首先进行可行性判断,判断其是否满足约束条件:①满足未被访问过;②满足车辆最大行驶距离限制;③满足车辆载重量限制。如果存在可行性节点,按照公式(2)和公式(3)选择移动到下一个节点 j ,并按照公式(3)进行信息素的局部更新,重复步骤3,否则转到步骤4。

步骤4 所有客户节点都已在当前解集中,计算目标函数值,转到步骤6,否则转到步骤5。

步骤5 重新开始一条新的路径并重复上述构造过程,如在寻找过程中无法找到可行的下一个客户时返回配送中心,然后再从配送中心继续出发直到所有的客户点都被访问。

步骤6 在可行解中选出本次迭代最优解 s ,以 s 为初始解,执行TS过程如图1所示。ACO算法得到的解 s 作为初始解,并置该解为当前最好解,禁忌表 K 置空。

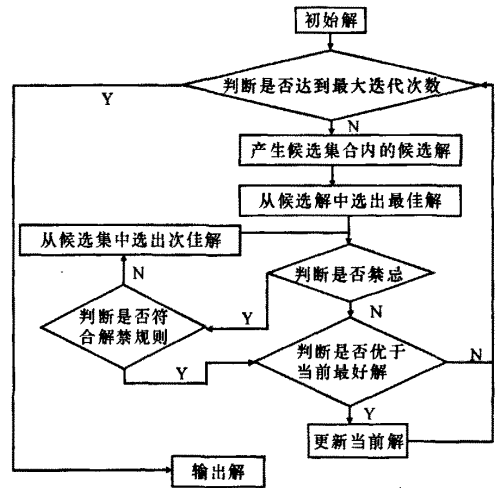


图1 TS算法流程

步骤7 按照公式(5)进行全局信息素更新。

步骤8 如果满足终止条件则结束;否则 $I=I+1$,转向步骤2。输出最优解。

3 数据实验

用Salhi和Nagy算例作为实验测试数据^[7]进行3组实验,以验证ACO-TS算法的性能。实验中的参数设置相同。表1中第一列是算例名称, N 是客户个数, k 表示车辆个数, $cost$ 表示每组实验中的最优解的费用, $time$ 表示CPU运行时间。

通过表1结果可以看出,ACO-TS算法求解质量优于单独使用ACO算法或者TS算法,虽然求解时间比单独使用TS算法要长,但是解的质量有所

提高,车辆使用数量上与 ACO 算法的 14 个算例相同,而比 TS 算法算例中的 2 个算例的车辆数少 1。这表明将 ACO 算法和 TS 算法结合是有效的,能在较短的时间取得较高质量的解。

表 1 3 种算法结果

Y	N	ACO			TS			ACO-TS		
		k	cost	time	k	cost	time	k	cost	time
CMT1X	50	3	515.56	0.56	3	503.22	0.08	3	450.63	1.20
CMT1Y	50	3	498.23	0.52	3	501.22	0.09	3	456.66	1.12
CMT2X	75	6	826.12	0.89	6	801.11	0.19	6	680.23	4.58
CMT2Y	75	6	801.69	0.96	6	799.86	0.20	6	667.02	4.80
CMT3X	100	5	852.36	3.22	5	799.11	2.19	5	702.89	12.01
CMT3Y	100	5	856.50	3.15	5	795.98	2.00	5	700.58	12.23
CMT4X	150	7	1109.45	70.10	8	989.56	2.15	7	842.12	42.13
CMT4Y	150	7	1205.22	7.23	8	978.56	3.00	7	831.79	43.00
CMT5X	199	10	1498.23	15.26	10	1123.35	4.12	10	1005.23	98.36
CMT5Y	199	10	1501.60	14.28	10	1100.29	4.23	10	1065.32	99.22
CMT11X	120	4	1023.66	10.10	4	899.36	1.50	4	826.53	21.55
CMT11Y-120	4	997.58	9.85	4	900.12	1.50	4	834.21	22.21	
CMT12X	100	5	819.69	3.17	5	780.56	2.05	5	689.12	10.23
CMT12Y	100	5	835.41	2.98	5	780.05	2.16	5	695.28	10.15
AVG		5.7	952.95	5.65	5.85	839.45	1.81	5.7	746.26	27.34

4 结束语

本文针对 VRSPDP 问题提出一种新的蚁群禁忌混合优化算法,该算法采用 ACO 算法产生的解作为 TS 算法的初始解,并且采用 MIN-MAX 思想对信息素进行限制,通过构造合理的邻域结构,选取合适的禁忌长度及终止规则提高解的质量及搜索速度。实验结果表明,新算法的优化质量和效率都优

于传统蚁群算法和禁忌算法。但是新算法没有考虑多站点、带时间窗的 VRSPDP 问题,这将有待我们进一步深入研究。

参考文献:

- [1] Min H. The multiple vehicle routing problem with simultaneous delivery and pick-up points [J]. Transportation Research A, 1989,23A : 377 - 386.
- [2] Fermin Alfredo Tang Montane,Roberto Dieguez Galvao. A tabu search for the vehicle routing problem with simultaneous pick-up and delivery[J]. Computers & Operations Research,2006,33:59-61.
- [3] Chen J F. Approaches for the vehicle routing problem with simultaneous deliveries and pickups[J]. Journal of the Chinese Institute of Industrial Engineers,2006, 23 (2):141-150.
- [4] Nagy G,Salhi S. Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries [J]. European Journal of Operational Research,2005,16:126-141.
- [5] Dorigo M,ManiezzoV,Colomi. A ant system;an autocatalytic optimizing process[R]. Technical Report, 91-106.
- [6] 段海滨. 蚁群算法原理及其应用[M]. 北京:科学出版社,2005:26-29.
- [7] Nagy G,Salhi S. A cluster insertion heuristic for single and multiple depot vehicle routing problem with backhauling [J]. Journal of the Operational Research Society,1999,50(10):1034-1042.

(责任编辑:韦廷宗)

以色列开发出激光焊接手术伤口技术

以色列特拉维夫大学亚伯拉罕·卡茨尔教授领导的研究小组开发出一种新的安全有效的手术伤口激光焊接技术,该技术使用一种特殊白蛋白作为生物黏合剂涂到手术切口两侧,使用激光加热后,生物黏合剂会形成一种“硬壳”,保护伤口不受感染并促其快速愈合。之前医学界也曾尝试用激光焊接技术缝合手术伤口,但是由于温度难以控制,容易因过热伤及人体组织。新的手术伤口激光焊接技术使用了可控温二氧化碳激光器和卤化银,使加热温度能有效地控制在适宜范围内,避免产生过热或灼伤等现象。

研究人员已对 10 位实施胆囊手术的患者进行临床试验并取得成功。与传统缝合技术相比,采用激光焊接技术,手术伤口愈合更快,效果更理想。卡茨尔教授表示,下一步他们计划在剖腹产等产生较长伤口的手术上进行试验,如果取得成功,预计几年内可投入商业使用。

(据科学网)