

基于在线捆绑的服务产品动态定价模型求解算法分析

杨清清, 欧朝敏, 郭滕达

(国防科学技术大学 信息系统与管理学院, 湖南 长沙 410073)

摘 要: 通过三个数量产品的捆绑选择, 证明了紧急补货模型 (ER) 和失销模型 (LS) 的动态捆绑问题的复杂性, 提出启发性算法求解动态定价和捆绑的决策问题. 通过算例比较了短视启发式算法、静态确定性近似算法、两阶段启发式算法和损耗率启发式算法分别在 ER 和 LS 模型中捆绑和定价决策的有效性和稳健性, 结果表明在产品数量不多时两阶段算法是最佳选择, 在产品数量较多或者顾客对产品捆绑价格敏感性较低时损耗率算法中的 DRO 算法更有效. 分析有助于选择合适的算法对捆绑包结构和捆绑包价格进行计算.

关键词: 动态定价; 在线捆绑; 服务产品; 算法分析

1 引言

服务产品包括航空客运、酒店业、汽车租赁、金融服务等, 具有价值易逝性, 市场可分性, 需求的波动性和供给能力短期非调节性等特点. 最显著的特点是服务的易逝性, 又被称为不可储存性, 指的是服务作为一种非实体的产品, 不管在时间上还是在空间上都是不可储存的. 首先, 服务不能在生产后储存待售, 例如飞机上的座位不能储存, 这趟航班剩下的座位是不可能保存到下一趟航班的. 其次, 服务客户也无法购买后储存, 例如一旦客户离开酒店, 酒店的服务也即消失, 无法再享受这样的服务. 许多服务产品实质上都是属于易逝品. 而产品具有易逝性是收益管理应用的一大条件. 产品的易逝性使得企业必须在产品失效之前卖出尽可能多的产品, 并最大化挖掘产品价值. 如何制定易逝性产品的价格、供给等决策, 以实现收入的增长, 也是收益管理适合且擅长发掘解决方案的一类管理问题.


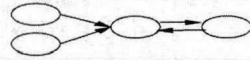
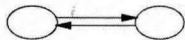


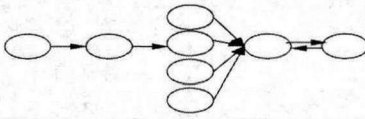

捆绑销售指销售商将一种产品与其他产品组合在一起以一个价格出售. 捆绑销售可以降低顾客支付意愿的差异性, 因此对服务产品销售企业来说, 在捆绑基础上实现歧视定价无疑是最佳途径之一. 例如, 从卓越网上购买书籍, 都会在“最佳组合”中获得一个购买包含该书的捆绑套系书籍的推荐. 下面将从 2014 年 6 月 10 日卓越网的“图书销售排行榜”选出的 100 本最畅销书名单来分析推荐捆绑销售的构造特征, 如表 1 所示. 另外, <http://www.e-tailing.com> 通过调查显示, 前 100 位的网络零售商中 72% 都运用了各种形式的捆绑销售策略. 旅游相关的网络公司, 航空公司和酒店都出现了各种运用捆绑销售的例子. 所有这些例子都强调了交易中捆绑销售决策的重要性, 并且迫切需要将这最优化决策进行定量化分析.


收稿日期: 2014-12-29


资助项目: 国家自然科学基金项目《网络环境下收益管理中服务产品动态定价研究》(71102004); 湖南省哲学社会科学基金项目《运输型物流企业中基于马尔科夫决策过程的收益管理 Robust 模型研究》(2010YBA022); 湖南省软科学研究计划项目《生产性服务业中基于顾客选择行为的动态定价策略研究》(2010zk3012)

电子商务环境下基于在线捆绑的服务产品动态定价模型是建立在大量收益管理动态定价研究成果基础上的^[1-6]. 在电子商务环境下执行捆绑销售带来了一系列挑战. 首先, 捆绑销售的实施依赖于顾客每天的购买信息而不是以前使用的静态规则. 捆绑销售通常提供一定折扣, 也就是说, 捆绑的产品会产生较低的边际利润. 因此, 如果捆绑包中某一产品的存量较低, 那么单独销售该产品可能更有利. 因此, 当前存量信息也必须考虑到捆绑销售决策中. Huiling Chung 等^[7]研究了不同产品的捆绑策略. Abraham 和 Thierno^[8]以有线信道为例, 探讨了信息产品的捆绑和定价问题. Michalel 和 Tuomas^[9]基于顾客历史购买数据提出了一个自动捆绑定价的框架, 并且开发了几种搜索算法, 确定利润最大化的价格和捆绑折扣. 杨清清等^[10]运用动态规划和组合优化理论, 建立了基于在线捆绑的易逝品动态标价模型; 针对是否补充缺货分辨建立紧急补货模型和失销模型.

表 1 卓越网 100 本畅销书的捆绑结构分析表

捆绑结构	观察数目	捆绑结构	观察数目
	51		6
	12		7
	14		6
	4		

 在销售排行榜100位里

 不在销售排行榜100位里

(资料来源: 本研究整理)

本文将在文献 [10] 的基础上, 分别针对紧急补货模型和失销模型给出一些启发式求解算法, 并比较算法的有效性和稳健性, 最后加以数值检验.

2 基于在线捆绑的服务产品动态标价模型的构建

针对紧急补货和失销两种状态分别构建基于在线捆绑的服务产品动态标价模型: 第一种模式中, 服务产品销售企业有机会以额外成本获得一件脱销产品, 称为紧急补货模型 (Emergency Replenishment, 简称 ER 模型); 在第二种模式中, 服务产品存量不能补充, 一旦产品脱销将拒绝顾客需求, 称为失销模型 (Lost-Sales Model, 简称为 LS 模型).

考虑服务产品销售企业在有限的时间内销售 m 组系列产品, 将其分割成 N 个时间决策单元, 在每个决策单元的初期, 服务产品卖方根据每个产品的库存做出捆绑和定价决策. 在电子商务环境下, 捆绑和定价决策需要进行更新, 因此相应的每个决策单元的时间长度比一般顾客到达间隔时间要短. 假定每个决策单元最多只有一个顾客到来. 用 $\lambda_i (i = 1, 2, \dots, m)$ 表示在任意决策单元, 第 i 类顾客到来, 并购买一个单位的第 i 组系列产品的概率, $\sum_{i=1}^m \lambda_i < 1$. 在第 i 类顾客提出以标价 p_i 购买产品 i 的请求后, 会收到服务产品卖方提供的“产品 i + 产品 j ”的捆绑包的购买建议, 其价格为 p_{ij} , 并假定服务产品卖方提供的捆绑包仅包含两个产

品. 第 i 类顾客对 ij 捆绑产品具有保留价格, 服从累积分布函数 $F_{ij}(\cdot) > 0$. 因此, 第 i 类顾客购买捆绑产品的概率为 $\bar{F}_{ij}(p_{ij}) = 1 - F_{ij}(p_{ij})$; 仅购买产品 i 的概率为 $F_{ij}(p_{ij})$. 假定对于任意非负常量 y , $\bar{F}_{ij}(x)(x - y)$ 是关于 x 的单峰函数. 让 k 表示顾客编号, 那么每一个顾客都有其唯一编号 k , $\lambda_i = \sum_k \lambda_i^k$, λ_i^k 表示顾客 k 到达并购买产品 i .^[10]

每个决策单元都需要做出相关决定: 对于产品 i , 哪种产品 j 与其绑定, 捆绑包怎样确定价格. 在第 n 个决策单元 ($n = 1, 2, \dots, N$) 的初期, 产品的存量水平设定为向量 $\mathbf{I} = (I_1, \dots, I_m)$, 令 $V_n(\mathbf{I})$ 表示从第 n 个决策单元到销售期结束的最优期望收益. 定义 \mathbf{e}_i ($i = 1, \dots, m$) 为一个 m 维向量空间, 如果 $k = i$, 则 $(\mathbf{e}_i)_k = 1$, 反之为 0. 企业的目标是实现 $V_1(\mathbf{I})$ 的最大化, $V_n(\mathbf{I})$ 依赖于实际存量水平, 如果每种产品至少有一个单位的存量, 即 $I_i \geq 1$ ($i = 1, \dots, m$), $V_n(\mathbf{I})$ 可以用贝尔曼方程表示为:

$$V_n(\mathbf{I}) = \sum_{i=1}^m \lambda_i \max_{j \neq i} \left(\max_{p_{ij}} (F_{ij}(p_{ij})(p_i + V_{n+1}(\mathbf{I} - \mathbf{e}_i)) + \bar{F}_{ij}(p_{ij})(p_{ij} + V_{n+1}(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j))) \right) + \left(1 - \sum_{i=1}^m \lambda_i \right) V_{n+1}(\mathbf{I}) \quad (1)$$

式 (1) 中有两个最大化算子能体现出动态捆绑和定价决策. 外部的最大化算子确定了在第 i 个顾客到来时与产品 i 捆绑的最佳的产品 j ; 里面的最大化算子确定了 ij 产品捆绑包的最优价格. 根据服务产品价值易逝的特性, 假定在销售期限结束后剩余的存量价值为零, 于是有附加条件 $V_{N+1}(\mathbf{I}) = 0$.

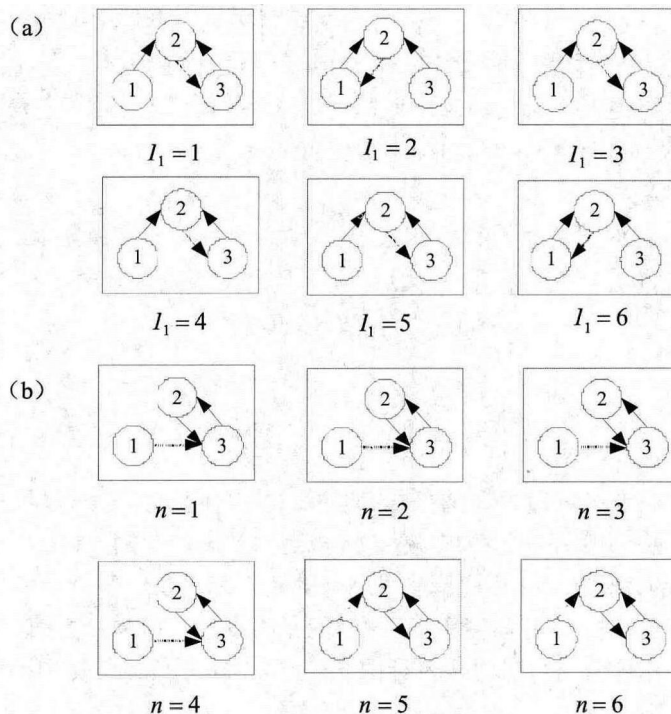


图 1 三个产品的情形: 最优动态捆绑是存量 ((a) $I_2 = 2, I_3 = 4, n = 1$) 和时间 ((b) $I_1 = 1, I_2 = 2, I_3 = 2$) 的函数. 参数设定为, $N = 10, \beta = 1, p_1 = p_3 = 1, p_2 = 1.5, \lambda_3 = 0.3, \bar{F}_{ij}(p_{ij}) = ((p_i + p_j - p_{ij})/p_j)^\beta$

无疑, 动态捆绑问题是很复杂的, 因为它需要解决大量随机动态规划的组合优化问题. 以三个数量产品为例来看此类问题的复杂性, 图 1 说明了在给定下列参数后, 三种产品情形下的最优捆绑策略: $\bar{F}_{ij}(p_{ij}) = ((p_i + p_j - p_{ij})/p_j)^\beta$, 定义在 $[p_i, p_i + p_j]$ 中, $i, j = 1, 2, 3$, $N = 10, \beta = 1, p_1 = p_3 = 1, p_2 = 1.5$, 并且 λ . 图 1a 表示了在销售期一开始 ($n = 1$), 最优捆绑策略怎样随着产品 1 的存量变化而变化: 作为捆绑补充产品 2, 在产品 1 和 3 之间转换; 图形 1b 体现的是同样情形下最优捆绑策略的动态性: 剩余时间的逐渐减少迫使产品 1 的捆绑补充由产品 3 向产品 2 转换. 说明了产品的捆绑决策是非常复杂的. 假设公式 (1) 中单一定价最优化问题的计算复杂性是恒定且不依赖于捆绑问题的复杂程度, 解决这一动态规划的组合优化问题需要 $O((N \times \prod_{i=1}^m I_i)^{m^2})$ 步操作. 捆绑和动态定价之间的交互显著地增加了动态规划问题的复杂性.

3 基于在线捆绑的服务产品动态定价模型求解算法分析

3.1 ER 模型下捆绑和动态定价的启发式求解算法及算例分析

当基于捆绑的服务产品动态定价问题有大量产品存在时, 寻求最优捆绑补充的组合优化问题面临巨大的挑战. 在这一部分, 针对 ER 模型比较一些捆绑和定价启发式算法的有效性和稳健性, 并进行数值算例验证. 首先, 提出短视启发式算法 (Myopic Heuristic) H^M , 这种算法忽略了产品存量和时间消逝. 之后考虑产品存量和时间因素, 提出两个更复杂的启发式算法: 静态确定性近似启发式算法 (Static Deterministic Approximation) H^D 是一个有静态固定需求, 并且价值依赖于捆绑和定价策略的启发式算法, 它能动态地在每个时点反应存量水平的变化; 两阶段启发式算法 (Two-Stage Heuristic) H^T 通过假设除了当前这个时期在其他任意时期都没有捆绑决策来简化动态规划模型的求解. 最后, 损耗率启发式算法 (Depletion Ratio Heuristic) H^{DR} 基于容易计算的编号来做出决策.

3.1.1 短视启发式算法

短视算法 H^M 忽视产品存量的影响, 是解决动态定价和捆绑最简单的方法. 根据 [10] 中公式 (9), 假定 $G_{n+1}^i(I_i) = G_{n+1}^i(I_i - 1)$, 可得:

$$j^M(i) = \arg \max_{j \neq i} \bar{F}_{ij}(p_{ij}^M)(p_{ij}^M - p_i), i = 1, \dots, m \quad (2)$$

其中

$$p_{ij}^M = \arg \max_{p_{ij}} (F_{ij}(p_{ij})(p_{ij} - p_i)) \quad (3)$$

由 (2) 定义的捆绑策略短视地选择在销售捆绑包时能最大化期望收益的捆绑补充. 公式 (2) 体现的是购买 i 产品的顾客在产品 j 选择上的倾向性, 并且提出了当产品 j 的存量是无限时, ij 捆绑包的最优价格. 短视算法最明显的优点就是执行和计算的简单: 他仅仅需要 $O(m^2)$ 次比较 (假定执行公式 (3) 所需要的时间是不变的). 但另一方面, 短视算法假设捆绑补充产品存量的边际价值是可以忽略不计的, 这一假设在产品存量受到约束的时候是非常不恰当的.

3.1.2 静态确定性近似算法

这部分将对动态随机的 ER 模型考虑一个静态确定近似方法 H^D . 假设对每个捆绑包和每单个产品的一次性需求是确定的, 但是受到捆绑和动态定价决策的影响. 引入一个时间参

数 $\tau = N - n$, 表示在静态确定性分析过程中的有效时间期限. 顾客对产品 i 的总需求为 $\lambda_i \tau$. 捆绑决策由产品 i 的需求定义, 即 $0 \leq q_{ij} \leq 1$, 这类顾客会被提供给一个 ij 捆绑包. 这些连续变量对这一确定性模型的分析有所简化.

给定捆绑决策 $\{q_{ij}\}$, 定价决策 $\{p_{ij}\}$, 对 ij 捆绑包的总需求为 $\lambda_i \tau q_{ij} \bar{F}_{ij}(p_{ij})$, 并且给定对单个产品 i 的总需求为 $\sum_{j \neq i} \lambda_i \tau q_{ij} F_{ij}(p_{ij})$, 其中 $i, j = 1, 2, \dots, m, i \neq j$. 服务产品下搜后企业的目标是选择捆绑决策 $\{q_{ij}\}$ 和定价决策 $\{p_{ij}\}$ 的最优值使得总收益减去总的紧急补货成本后的值最大化. 这一静态问题 (P1) 可以模型化为:

$$\begin{aligned}
 (P1) \quad & \max_{q_{ij}, p_{ij}} \sum_{i=1}^m \sum_{j \neq i} \lambda_i \tau q_{ij} \bar{F}_{ij}(p_{ij}) p_{ij} + \sum_{i=1}^m \sum_{j \neq i} \lambda_i \tau q_{ij} F_{ij}(p_{ij}) p_i \\
 & - \sum_{i=1}^m b_i \left(\lambda_i \tau + \sum_{j \neq i} \lambda_j \tau q_{ji} \bar{F}_{ji}(p_{ji}) - I_i \right)^+ \\
 \text{s.t.} \quad & \sum_{j \neq i} q_{ij} = 1, \quad i = 1, 2, \dots, m \\
 & q_{ij} > 0, i \neq j
 \end{aligned}$$

其中 $x^+ = \max\{x, 0\}$. 模型目标函数的第一部分表示的是销售所有的 ij 捆绑包所得到的总收益, 第二部分是出售单个产品的总收益, 第三部分是紧急补货模型的采购成本. 对产品 i 的总需求是由对单个产品 i 的需求 $\sum_{j \neq i} \lambda_i \tau q_{ij} F_{ij}(p_{ij})$, 对 ij 捆绑包的需求 $\sum_{j \neq i} \lambda_i \tau q_{ij} \bar{F}_{ij}(p_{ij})$ 和对 ji 捆绑包的需求 $\sum_{j \neq i} \lambda_j \tau q_{ji} \bar{F}_{ji}(p_{ji})$ 组成.

由于模型 (P1) 属于带有约束的最优化问题, 其目标函数是不可微的, 一般来说, 这类问题是难以求解的. 为使得模型容易求解, 下一步将提出一个等价于 (P1) 模型的改进模型. 首先, 构造一个对偶问题 (D2), 并且证明 (P2) 的最优解法可以很容易从 (D2) 的解法中获得. 由于 (D2) 的目标函数是线性的, 约束集是凸集, 现有的很多算法, 例如梯度算法, 都适用于 (D2) 的求解, 这样对模型 (P1) 的求解就容易多了.

引入辅助变量 $\{y_i\}$ 并且对 (P1) 的目标函数进行一些代数运算, (P1) 模型可以转化为 (P2):

$$\begin{aligned}
 (P2) \quad & \max_{y_i, q_{ij}, p_{ij}} \sum_{i=1}^m \sum_{j \neq i} \lambda_j \tau q_{ji} \bar{F}_{ji}(p_{ji}) (p_{ji} - p_j) - \sum_{i=1}^m b_i y_i \\
 \text{s.t.} \quad & y_i \geq \lambda_i \tau + \sum_{j \neq i} \lambda_j \tau q_{ji} \bar{F}_{ji}(p_{ji}) - I_i \\
 & \sum_{j \neq i} q_{ij} = 1, i = 1, 2, \dots, m \\
 & y_i \geq 0, i = 1, 2, \dots, m \\
 & q_{ij} \geq 0, i \neq j
 \end{aligned}$$

定义 $H_{ji}(x) = \max_{p_{ji}} (\bar{F}_{ji}(p_{ji}) (p_{ji} - p_j - x))$. 这部分的分析结论总结于命题 1.

命题 1 a) 根据模型 (P2) 可以构造对偶问题 (D2), 并模型化为下列最优化问题, 其约束集是凸集.

$$(D2) \quad \min_{\mu_i, v_i} \sum_{i=1}^m (\mu_i (I_i - \lambda_i \tau) + v_i)$$

$$s.t. \lambda_j \tau H_{ji}(\mu_i) \leq v_j, i \neq j$$

$$0 \leq \mu_i \leq b_i, i = 1, 2, \dots, m$$

b) 给定 (D2) 的任意最优解 $\{\mu_i^*, v_i^*\}$, (P2) 存在对应的最优解 $\{\{p_{ij}^*\}, \{q_{ij}^*\}, \{y_i^*\}\}$, 并且有如下性质: i) $p_{ji}^* = \arg \max_{p_{ji}} (\bar{F}_{ji}(p_{ji})(p_{ji} - p_j - \mu_i^*))$, $i \neq j$; ii) 以 p_{ij}^* 替代 p_{ij} , $\{q_{ij}^*\}$ 和 $\{y_i^*\}$ 是 (P2) 的解.

命题 1b i) 表明对偶解 μ_i^* 可以理解为额外增加一个单位的产品 i 的边际价值. 如果货源充足, 没有必要对产品 i 进行紧急补货, 那么 $\mu_i^* = 0$; 相反, 如果货源不足, 那么 μ_i^* 等于紧急补货成本 b_i . 由于 (P1) 和 (P2) 等同, 命题 1b 得到的最优解 $\{p_{ij}^*, q_{ij}^*\}$ 对于模型 (P1) 同样是最优的.

算法 H^D 执行步骤如下: 在每个决策单元 n 的初始, 使用参数 $\tau = N - n$ 和当前产品存量水平来求解 (P2). 所得的捆绑策略 ($\{q_{ij}^*\}$) 和定价策略 ($\{p_{ij}^*\}$) 有如下用途: 当购买产品 i 的顾客到达, 企业进行随机的“投掷硬币”试验 (结果为 j 的概率是 q_{ij}^*); 选择 j 为捆绑补充, ij 捆绑包以价格 p_{ij}^* 被提供给顾客进行选择. 这一算法包括解决 $O(N)$ 个凸优化问题 (D2) 和线性规划问题 (P2).

3.1.3 两阶段启发式算法

第 n 个决策单元的初始存量设为 \mathbf{I} . 在两阶段法中, 为了简化捆绑和动态定价问题, 假设在 $n+1$ 期间没有捆绑. 在每个决策单元, 购买产品 i 的顾客到来的数量是一个参数为 λ_i 的伯努利随机变量; 在剩余决策单元 $N - n$, 购买产品 i 的顾客到来的数量是一个参数为 $(\lambda_i, N - n)$ 的二项式随机变量 Λ_i . 增加一个单位产品 $j \neq i$ 的边际价值可以表示为:

$$\begin{aligned} V_{n+1}(\mathbf{I} - \mathbf{e}_i) - V_{n+1}(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j) &= b_j E(\min(\Lambda_j, I_j) - \min(\Lambda_j, I_j - 1)) \\ &= b_j \Pr(\Lambda_j \geq I_j) = b_j \sum_{k=I_j}^{N-n} \binom{N-n}{k} \lambda_j^k (1 - \lambda_j)^{N-n-k} \end{aligned} \quad (4)$$

任意 (i, j) 组合的捆绑价格 p_{ij}^T 可以从式 (4) 中获得:

$$\begin{aligned} \partial V_n(\mathbf{I}) / \partial p_{ij} &= f_{ij}(p_{ij})(p_i + V_{n+1}(\mathbf{I} - \mathbf{e}_i)) - f_{ij}(p_{ij})(p_{ij} + V_{n+1}(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j)) + \bar{F}_{ij}(p_{ij}) \\ &= -f_{ij}(p_{ij})(p_{ij} - p_i - (V_{n+1}(\mathbf{I} - \mathbf{e}_i) - V_{n+1}(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j))) + \bar{F}_{ij}(p_{ij}) = 0 \end{aligned} \quad (5)$$

将 (4) 代入 (5) 整理可得:

$$p_{ij}^T = p_i + \bar{F}_{ij}(p_{ij}^T) / f_{ij}(p_{ij}^T) + b_j \sum_{k=I_j}^{N-n} \binom{N-n}{k} \lambda_j^k (1 - \lambda_j)^{N-n-k} \quad (6)$$

短视价格 p_{ij}^M 的表达式不包括公式 (6) 中最后一项, p_{ij}^T 是 p_{ij}^M 的上限值, 由于不考虑后续时期的捆绑策略, p_{ij}^T 也是最优价格的下限值. p_{ij}^T 明显依赖于产品 j 的存量.

假定顾客 i 到来而不提供任何捆绑, 这样, 可获得收益 $p_i + V_{n+1}(\mathbf{I} - \mathbf{e}_i)$, 提供 ij 捆绑增加的收益 Δ_{ij} , 可由下式计算得出:

$$\begin{aligned} \Delta_{ij} &= (F_{ij}(p_{ij}^T)(p_i + V_{n+1}(\mathbf{I} - \mathbf{e}_i)) + \bar{F}_{ij}(p_{ij}^T) \\ &\quad \cdot (p_{ij}^T + V_{n+1}(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j))) - (p_i + V_{n+1}(\mathbf{I} - \mathbf{e}_i)) \\ &= \bar{F}_{ij}(p_{ij}^T)(p_{ij}^T - p_i - (V_{n+1}(\mathbf{I} - \mathbf{e}_i) - V_{n+1}(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j))) \\ &= \bar{F}_{ij}(p_{ij}^T) \left(p_{ij}^T - p_i - b_j \sum_{k=I_j}^{N-n} \binom{N-n}{k} \lambda_j^k (1 - \lambda_j)^{N-n-k} \right) \end{aligned}$$

$$=(\bar{F}_{ij}(p_{ij}^T))^2 / f_{ij}(p_{ij}^T)$$

(由公式(6)得).

算法 \mathbf{H}^T 执行步骤如下: 当第 i 类型顾客到来时, 对所有产品 j 使用公式(6) 计算 p_{ij}^T , 接着算出 Δ_{ij} 的值, 根据(7) 选出最大值:

$$j^T(i) = \arg \max_{j \neq i} \left((\bar{F}_{ij}(p_{ij}^T))^2 / f_{ij}(p_{ij}^T) \right) \quad (7)$$

在两阶段启发式算法中, 捆绑和定价决策都是动态的: 他们都依赖于当前产品存量和时间消逝. 两阶段算法的计算复杂度为 $O(Nm^2)$.

举例继续说明 \mathbf{H}^T 算法的捆绑决策, 设定对称的价格敏感性因素为 β_{ij} , 仅依赖于第一选择产品编号即 $\beta_{ij} = \beta_i$, 价格保留函数为 $(\bar{F}_{ij}(p_{ij}) = \exp(-\beta_{ij}(p_{ij} - p_i)))$, 对称额外成本 $b_j = b$, 可得:

$$p_{ij}^T = p_i + \frac{1}{\beta_i} + b \sum_{k=I_j}^{N-n} \binom{N-n}{k} \lambda_j^k (1 - \lambda_j)^{N-n-k} \quad (8)$$

进行一些代数变化后, 得:

$$j^T(i) = \arg \max_{j \neq i} \left(\sum_{k=0}^{I_j} \binom{N-n}{k} \lambda_j^k (1 - \lambda_j)^{N-n-k} \right) \quad (9)$$

引入 $S(n, I_j, \lambda_j)$,

$$S(n, I_j, \lambda_j) = \sum_{k=0}^{I_j} \binom{N-n}{k} \lambda_j^k (1 - \lambda_j)^{N-n-k} \quad (10)$$

$S(n, I_j, \lambda_j)$ 是关于产品存量 I_j 的递增函数, 决策单元 n 是需求强度 λ_j 的递减函数, 并且不依赖于第一选择产品 i 的编号. 也就是说, $S(n, I_j, \lambda_j)$ 可以看成是在决策单元 n 产品 j 销售情况的指示值: 其值越低, 产品销售状况越佳. 因此, ij 捆绑包的价格是 I_j 和决策单元 n 的递减函数, 是需求强度 λ_j 的递增函数. 捆绑补充可由以下步骤得到: 在每个决策单元 n , 将所有产品按照当前参数 $S(n, I_j, \lambda_j)$ 的值进行排序, 令 $\hat{j}_s^T = \arg \max_j (S(n, I_j, \lambda_j))$, $\hat{j}_n^T = \arg \max_{j \neq \hat{j}_s^T} (S(n, I_j, \lambda_j))$ 分别表示具有最高参数值和第二高参数值产品的编号, 公式(9) 可以等价表示为:

$$j^T(i) = \begin{cases} \hat{j}_s^T, & i \neq \hat{j}_s^T \\ \hat{j}_n^T, & i = \hat{j}_s^T \end{cases} \quad (11)$$

由于将 $S(n, I_j, \lambda_j)$ 看成是最佳销售的编号, 公式(11) 确定的捆绑策略就很直观了: 所有产品都将与当前销售最差的产品进行捆绑, 销售最差的产品本身将与当前销售第二差的产品进行捆绑.

3.1.4 损耗率启发式算法

对两阶段启发式算法的探讨发现加入捆绑策略对于通过畅销产品的销售带动滞销产品的销售是一个有效的方法. 下面将提出另一种容易执行的算法——损耗率启发式算法 \mathbf{H}^{DR} , 它强调动态捆绑在服务产品销售过程中的重要作用. \mathbf{H}^{DR} 算法指的是: 在每个决策单元 n , 设定产品 i 当前的产品存量为 I_i , 给每个产品 i 设定一个“损耗率”编号, 等于 I_i 与 λ_i 的比值. 通过定义不难发现, 损耗率编号与两阶段算法中 $S(n, I_j, \lambda_j)$ 起了相同的作用: 他对每个

产品当前销售情况进行了排序. 令 $j^s = \arg \max_i (I_i/\lambda_i)$ 和 $j^n = \arg \max_{i \neq j^s} (I_i/\lambda_i)$ 分别表示销售最差和销售第二差的产品, 那么损耗率算法中的动态捆绑策略如公式 (12) 所示:

$$j^{DR}(i) = \begin{cases} j^s, i \neq j^s \\ j^n, i = j^s \end{cases} \quad (12)$$

公式 (12) 所确定的捆绑补充的选择和公式 (11) 类似: 每个产品都将与当前销售最差的产品捆绑, 而同时, 销售最差的产品其本身将会与当前销售第二差的产品进行捆绑. DR 法没有限制捆绑定价策略的选择, 既可以与简单静态定价结合, 也可以与复杂一些的最优动态定价法相结合. 在下面的分析中, 基于 DR 算法考虑两个策略. 第一个策略将 DR 捆绑策略与 (3) 定义的短视定价策略结合, 称为短视 DR 策略 \mathbf{H}^{DRM} . 在 DRM 算法中, 对于每个决策单元 n , 产品 i 都将与产品 $j^{DR}(i)$ 捆绑, 捆绑包的价格为 $p_i^{DRM} = \arg \max_p (\bar{F}_{ij^{DR}(i)}(p)(p - p_i))$. 第二个基于 DR 算法的策略 \mathbf{H}^{DRO} 在 DR 捆绑策略下其捆绑补充由公式 (12) 决定. 也就是说, DRO 算法结合了 DR 捆绑策略和由下面动态规划所决定的定价策略:

$$V_n(\mathbf{I}) = \sum_{i=1}^m \lambda_i \max_p (F_{ij^{DR}(i)}(p)(p_i + V_{n+1}(\mathbf{I} - \mathbf{e}_i)) + \bar{F}_{ij^{DR}(i)}(p)(p + V_{n+1}(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j))) + \left(1 - \sum_{i=1}^m \lambda_i\right) V_{n+1}(\mathbf{I}) \quad (13)$$

\mathbf{H}^{DRM} 定价和捆绑补充决策的计算复杂度是 $O(Nm^2)$, 而 \mathbf{H}^{DRO} 的计算复杂度是 $O((N \times \prod_{i=1}^m I_i)^m)$.

尽管之前对 DR 捆绑策略有直观的分析, 但其达到最优的充分条件还是很难得到的, 除非在一些特殊限制性情形中. 下面的命题举出一个具备这样限制性条件的例子.

命题 2 令 $\lambda_i = \lambda, p_i = p, b_i = b, p_{ij} = q, p < q < p + b$, 并且 $\bar{F}_{ij}(p_{ij}) = \gamma, j \neq i; i, j = 1, 2, \dots, m$. 那么, 对于任意决策单元 $n = 1, \dots, N$, DR 捆绑策略都是最优的.

命题 2 考虑的是产品捆绑包以一个固定的价格销售, 并且产品仅在他们的存量价值上有所不同. 命题 2 揭示了 DR 捆绑策略在捆绑定价近似对称情形下的潜在有效性, 其在更多情形下的性能有效性还需要进行数值算例的检验.

3.1.5 算法的有效性比较及算例分析

这一部分将使用数值算例来检验 ER 模型中动态定价和捆绑启发式算法的有效性. R_{OPT} 表示对文献 [10] 中公式 (2) 求解所得的最优期望收益. 对每个启发式算法 π , 将根据下面的方法计算其期望收益值. 在每个决策单元 n , 让 $j^\pi(n, i)$ 表示根据启发性算法 π 为产品 i 选择的捆绑补充, $p^\pi(n, i)$ 表示 $i - j^\pi(n, i)$ 捆绑包的价格. 那么, 在任意初始存量下, 算法 π 的期望收益值可以根据式 (14) 求得:

$$V_n^\pi(\mathbf{I}) = \sum_{i=1}^m \lambda_i (F_{ij^\pi(n, i)}(p^\pi(n, i))(p_i + V_{n+1}^\pi(\mathbf{I} - \mathbf{e}_i)) + \bar{F}_{ij^\pi(n, i)}(p^\pi(n, i))(p^\pi(n, i) + V_{n+1}^\pi(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j)) + (1 - \sum_{i=1}^m \lambda_i) V_{n+1}^\pi(\mathbf{I})) \quad (14)$$

通过比较 $R_M, R_D, R_T, R_{DRM}, R_{DRO}$ 和 $R_{OPT} : \varepsilon_K = 100\% \times (R_{OPT} - R_k)/R_{OPT}, k = M, D, T, DRM, DRO$ 之间的相对性能差异来探究 ER 模型中算法 $\mathbf{H}^M, \mathbf{H}^D, \mathbf{H}^T, \mathbf{H}^{DRM}$ 和 \mathbf{H}^{DRO} 的有效性.

1) 产品数 $m = 3$, 价格保留函数为指数函数.

本章最开始对卓越网的观察发现,在大多数情况下通过捆绑联系的产品数目最多不超过3个(见表1),所以,以产品数量 $m = 3$ 为例进行分析.

数值算例的检验设计如下.为了隔离捆绑定价的影响,我们将单个产品的价格设定在同一个水平: $p_1 = p_2 = p_3 = 1$. 使用指数分布函数 $\bar{F}_{ij}(p_{ij}) = \exp(-\beta_{ij}(p_{ij} - p_i))$, 假设价格灵敏度因素 β_{ij} 仅仅依赖于第一选择产品的编号,即 $\beta_i = \beta_{ij}$. 价格灵敏度因素的取值有下列涵义: 1 代表低灵敏度, 2 代表中度灵敏度, 5 代表高灵敏度, 20 代表非常高的灵敏度. 决策单元数 $N = 20$, 总的顾客到达率为 $\lambda = \lambda_1 + \lambda_2 + \lambda_3 = 0.8$. 单个产品的顾客到达率设定为: $\lambda_1 = 0.1 + 0.5\sigma_1$, $\lambda_2 = 0.1 + (0.6 - \lambda_1)\sigma_2$, λ_3 式中的 σ_1 和 σ_2 取值为 0, 0.5, 1. 因此,到达率有七种可能的组合: $\{0.1, 0.1, 0.6\}$, $\{0.1, 0.6, 0.1\}$, $\{0.6, 0.1, 0.1\}$, $\{0.1, 0.35, 0.35\}$, $\{0.35, 0.1, 0.35\}$, $\{0.35, 0.35, 0.1\}$ 和 $\{0.35, 0.225, 0.225\}$. 这些组合包含了三种需求情形: 需求集中到单个产品, 需求集中到两个产品, 三个产品需求值相近.

为了检验算法对产品存量初始水平的灵敏度,我们将研究下列情形: 初始存量 $I_i = (1 + \gamma)\lambda_i N$, $i = 1, 2, 3$. 存量有效性系数 γ 的取值有下列涵义: -0.8 代表严格受限存量, -0.3 代表中等受限存量, 0 代表存量与需求相匹配, 0.3 代表适度松弛存量, 0.8 代表非受限存量. 对于 $i = 1, 2, 3$, 设定紧急补货价格 b_i 等于 ηp_i , η 的取值为 0.2, 0.5 和 0.8. 综合表 3-2 分析, $\varepsilon_M, \varepsilon_D, \varepsilon_T, \varepsilon_{DRM}$ 和 ε_{DRO} 的平均值分别为 10.63%, 3.16%, 0.12%, 7.75% 和 0.14%. 可以观察到两阶段算法和 DRO 算法在几乎所有的例子中执行得最为出色. 它们在所有测试中最差实例的相对性能差异也仅仅为 0.69%(两阶段算法) 和 1.04%(DRO 算法). 由于算法 H^D 在捆绑和定价决策中动态的结合了存量的影响, 确定性近似算法 H^D 表现得比短视算法 H^M 好, 在平均值上有 7.5% 的提高. DRM 算法的性能处于短视算法和确定性算法之间. 这些观察结果还强调了捆绑和定价策略匹配的重要性: 当损耗率捆绑策略与其最佳匹配的定价策略相结合时, 其算法性能近似最优, 而当其与短视定价策略结合时就失去了应有的效用.

表 2 ER 模型下算法关于 ER 价格函数的平均性能差异 (%)

η	0.2	0.5	0.8
ε_T	0.05	0.13	0.18
ε_D	0.91	2.74	5.83
ε_M	2.63	9.38	19.87
ε_{DRM}	1.61	6.72	14.92
ε_{DRO}	0.08	0.15	0.20

接下来分析算法关于 ER 额外费用大小 η (表 2)、存量有效性系数 γ (表 3) 和总的顾客需求(表 4)的灵敏度. 这些表格中 ε_i 值是这些数值算例的平均值.

H^T 和 H^{DRO} 启发性算法在许多情况下都有近乎最优的性能. 观察表 2 发现, H^M, H^D 和 H^{DRM} 在 η 值很小时性能较好. 表 3 表明, 当产品存量水平很高时, 产品的边际价值很小, 并且以上五个启发性算法都有较好的性能. 正如 (6) 和命题 1b 结论所示, 在高存量水平下, 两阶段、确定性算法和短视算法成为最优的; 另一方面, 当存量水平较低时, 产品 i 的边际价值接近于 b_i , 假设这种边际价值为零的短视算法的性能就变差了. 于是, 在低存量限制下, 两

阶段和确定性启发性算法决定的定价和捆绑决策近似最优. 最后, 表 4 表明对于总需求的变化, 五个启发性算法的性能不是很敏感: 两阶段和 DRO 算法仍旧为最佳选择, 其次是确定性近似算法, 接下来是 DRM 算法, 最后才是短视算法.

表 3 ER 模型下算法关于存量有效性的平均性能差异 (%)

γ	-0.8	-0.3	0	0.3	0.8
ε_T	0.03	0.14	0.20	0.15	0.08
ε_D	0.16	2.40	7.44	3.97	1.84
ε_M	18.47	11.50	10.18	7.50	5.48
ε_{DRM}	17.44	9.87	6.56	3.44	1.44
ε_{DRO}	0.00	0.01	0.08	0.35	0.26

表 4 ER 模型下算法关于到达率 λ_2 的平均性能差异 (%)

λ_2	0.1	0.225	0.35	0.6
ε_T	0.14	0.07	0.08	0.16
ε_D	3.52	2.47	2.27	3.82
ε_M	11.45	9.67	9.12	10.48
ε_{DRM}	7.97	7.28	7.22	8.15
ε_{DRO}	0.15	0.03	0.16	0.15

计算的复杂度从另外一个角度提供了一种可供算法进行比较的维度 (见表 5). 正如前面所分析的, H^M, H^D, H^T 和 H^{DRM} 算法有效性都较好, 相反, 像 H^{DRO} 这样最优算法的复杂度随着产品数目 m 呈指数增长, 并且是期初存量水平的多项式. 在数值测试中, 对所有算法性能的测评都被限制在 N 和 m 值相对较小的范围内.

表 5 ER 模型下算法计算复杂度

	操作步骤	(D2) 和 (P2) 最优化操作步骤
最优算法	$O\left((N \times \prod_{i=1}^m I_i)^{m^2}\right)$	
M	$O(m^2)$	
D		$O(N)$
T	$O(Nm^2)$	
DRM	$O(Nm^2)$	
DRO	$O\left((N \times \prod_{i=1}^m I_i)^m\right)$	

2) 检验两种最优算法的稳健性: 两阶段算法和损耗率算法

在上面所介绍的数值算例的研究中, 通过大量的参数测验, 认为两阶段算法和损耗率中的 DRO 算法具有最佳性能. 下面这部分分析的焦点将集中于探究两种算法关于价格灵敏度因素, 产品数量的变化和价格保留函数形式变化的稳健性.

首先考虑价格灵敏度因素 β_{ij} 不对称的情况. 在数值中引入下列变化: 允许系数 β_{13} 取值为 5, 10 和 20, 其他系数 $\beta_{ij} = 5$. 两种算法的性能总结在表 6 中, 结果发现两种算法在价格敏感性因素不对称时也有较好的性能. 但是, 当参数不对称加剧时两种算法的性能也有所

下降, 但是 DRO 算法仍强于两阶段算法。

表 6 ER 模型下算法关于不同的价格灵敏度因素的平均性能差异 (%)

β_{13}	5	10	20
ε_T	0.11	0.41	0.58
ε_{DRO}	0.14	0.25	0.36

接下来分析算法对于产品数量变化和价格保留函数形式变化的稳健性。为了研究价格保留函数形式变化的影响, 在指数保留函数 $\bar{F}_{ij}(p_{ij}) = \exp(-\beta_{ij}(p_{ij} - p_i))$ 之外, 还考虑保留函数为幂函数的情况 $\bar{F}_{ij}(p_{ij}) = ((p_i + p_j - p_{ij})/p_j)^{\beta_{ij}}$, 考虑从凸性的指数函数到凹性的幂函数对两种最优算法性能的影响。对两种形式的价格保留函数, 都使用对称性假设 $\beta_{ij} = \beta_i$, 并且设定 β_i 的取值为 0.5, 1.5。另外, 限制存量有效性系数 γ 的取值为 -0.3, 0, 0.3。在产品数量 $m = 3$ 时, 使用上面所描述的同样的需求模式; 在产品数量 $m = 4$ 时, 设定顾客总到达率为 $\lambda = \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 0.8$ 。单个产品的到达率分别为: $\lambda_1 = 0.1 + 0.4\sigma_1$, $\lambda_2 = 0.1 + (0.5 - \lambda_1)\sigma_2$, $\lambda_3 = 0.1 + (0.6 - \lambda_1 - \lambda_2)\sigma_3$, $\lambda_4 = 0.8 - \lambda_1 - \lambda_2 - \lambda_3$, 式中, σ_1, σ_2 和 σ_3 的取值分别为 0, 0.5 和 1。同之前的数值分析一样, 有 27 种可能的到达率组合, 涵盖了四种可能的情景: 主要为单个产品的需求, 主要为两个产品的需求, 主要为三个产品的需求和四种产品需求值相近。剩余参数的设定与之前数值算例所用的参数设定相同。所有数值算例测试结果都体现在表 7 中。

表 7 ER 模型下两阶段算法和 DRO 算法的平均 (最大) 性能差异 (%)

两阶段算法	指数保留函数	幂保留函数
3 个产品 4 个产品	0.13(0.69) 0.86(9.05)	0.24(6.98) 1.09(7.08)
DRO 算法	指数保留函数	幂保留函数
3 个产品 4 个产品	0.16(1.08) 0.46(3.01)	0.16(1.94) 0.25(2.29)

由于例中测试参数的某些变化, 使得同样在产品数量为 3, 价格保留函数为指数函数的情况下的分析结果与前面部分分析的结果有所不同。但是总的来说, 两种算法对于价格保留函数形式和产品数量的变化都体现了很高的稳健性。这一结论对 DRO 算法尤为有效, 他一直保持着近似最优的性能, 即使在最差的实例情形中, 相反, 在最差的实例情形中, 两阶段算法逐渐失去了其有效性。可以看到两阶段算法性能最差的实例是 $\beta_i = 0.5, \eta = 0.8$ 时, 即 1) 顾客愿意接受高的捆绑价格 (对于两种价格保留函数形式); 2) 脱销产品的补货成本很高昂。通过分析可以直观感觉到在捆绑销售被顾客接受的情况下, 忽略捆绑销售带来的机会的算法不会有很好的性能, 高的补货成本进一步强调了使用无效的捆绑方法会造成较大的收益损失。

DRO 和两阶段算法的选择可能最终依赖于服务产品销售企业运作过程中所面临的商业环境的特征。一方面, 如果企业存量的特性允许限制每个产品可能的产品补充数目最多为 2 个, 只要其顾客对于捆绑价格相对敏感, 那么就建议使用两阶段方法。因为 DRO 算法在每个决策单元都需要解决大量的最优化问题, 两阶段算法下的动态定价策略比 DRO 算法下的更容易计算。另一方面, 在产品数量较多或者顾客对产品捆绑价格的敏感性较低时, DRO 算法更有效。

3.2 LS 模型下捆绑和动态定价的启发式求解算法及算例分析

3.2.1 LS 模型下捆绑和动态定价的启发式求解算法

因为在 LS 模型下的动态定价问题也是很难处理的, 所以如果不使用启发性算法很难有效的解决动态定价和捆绑的决策问题. 下面将先分析 LS 模型中期望收益的上下限.

定义 1 设定 $L_n^i(I)$ 是满足下列递归关系的函数:

$$L_n^i(I) = \lambda_i (p_i + L_{n+1}^i(I-1)) + (1 - \lambda_i) L_{n+1}^i(I) \quad (15)$$

其中 $I \geq 1$, 并且对于 $i = 1, \dots, m, n = 1, \dots, N$, 当 $L_{N+1}^i(I) = 0$ 时, 有 $L_n^i(0) = 0$.

定义 2 设定 $U_n^i(I)$ 是满足下列递归关系的函数:

$$U_n^i(I) = \lambda_i \left(p_i + U_{n+1}^i(I-1) + \bar{F}_{i,j(i)} \left(p_{i,j(i)}^M \right) \left(p_{i,j(i)}^M - p_i \right) \right) + (1 - \lambda_i) U_{n+1}^i(I) \quad (16)$$

其中 $I \geq 1$, 并且对于 $i = 1, \dots, m, n = 1, \dots, N$, 当 $U_{N+1}^i(I) = 0$ 时, 有 $U_n^i(0) = 0$.

令 $L_n(\mathbf{I}) = \sum_{i=1}^m L_n^i(I_i)$; $U_n(\mathbf{I}) = \sum_{i=1}^m U_n^i(I_i)$. $L_n(\mathbf{I})$ 是没有捆绑的 LS 模型下的期望收益. 下面的命题指出 $L_n(\mathbf{I})$ 和 $U_n(\mathbf{I})$ 分别是 LS 模型中最优期望收益 $V_n^{LS}(\mathbf{I})$ 的下限和上限.

命题 3 a) 对于 $n = 1, \dots, N$, 有 $L_n(\mathbf{I}) \leq V_n^{LS}(\mathbf{I}) \leq U_n(\mathbf{I})$.

b) 对于 $n = 1, \dots, N$, 有

$$\frac{V_n^{LS}(\mathbf{I}) - L_n(\mathbf{I})}{L_n(\mathbf{I})} \leq \max \left\{ \frac{p_{j(i)}}{p_i} \mid i = 1, \dots, m \right\} \quad (17)$$

命题 3b) 中给出的上限是独立于顾客保留价格和顾客到达过程的. 如果每个产品的价格都远远超过其捆绑补充, 那么 (17) 设定的限制将会很严格, 表明捆绑决策对于预期收益的影响相对于无捆绑时的情况是可以忽略的. 给定每个产品的价格, 就容易计算出每种捆绑策略的上限. 因此, 在没有需求信息的情况下, 根据严格上限仍然能够确定捆绑结构. 看一个例子, 例中 $(p_1, p_2, p_3) = (100, 10, 1)$, 产品 1 与产品 2 相捆绑, 产品 2 与产品 3 相捆绑, 产品 3 不与其他任何产品进行捆绑. 根据公式 (17), 在这一捆绑结构下, 期望收益最多可以超过没有加入捆绑销售情况的 10%.

根据文献 [10] 中公式 (3.20), 求解最优捆绑价格可以表示为下列最大化问题:

$$p_{ij}^*(\mathbf{I}, n) = \arg \max_{p_{ij}} (\bar{F}_{ij}(p_{ij}) (p_{ij} - p_i + (V_{n+1}(\mathbf{I} - \mathbf{e}_i - \mathbf{e}_j) - V_{n+1}(\mathbf{I} - \mathbf{e}_i)))) \quad (18)$$

对 (18) 的一个近似改进是使用 (3) 定义的捆绑价格 p_{ij}^M , 忽略 (18) 的后两项. 这一启发式捆绑价格在时间期限内是恒定的并且独立于任意产品的存量水平. 仍然用 \mathbf{H}^M 表示这一启发性算法. 下面结论表明 p_{ij}^M 是最优价格 $p_{ij}^*(\mathbf{I}, n)$ 的下限, 并分析了启发性算法 \mathbf{H}^M 是最优的充分条件:

命题 4 a) 最优价格不会小于由启发性算法 \mathbf{H}^M 所决定的价格, 即对于任意存量 \mathbf{I} 和决策单元 n , 都有 $p_{ij}^*(\mathbf{I}, n) \geq p_{ij}^M$.

b) 当可提供的存量 $I_k \geq N + 2 - n, k = 1, 2, \dots, m$ 时, 启发性算法 \mathbf{H}^M 在第 n 个决策单元为最优.

命题 4a) 的证明利用了命题 3.4 提出的价值函数的单调性. 命题 4b) 指出当每个产品的存量水平都很高的时候短视启发式算法是最优的. 当存量受到更多限制时, 短视算法的性能有效性会变差.

所有在 ER 模型中使用的启发性算法都能修改而适合 LS 模型. 对于 H^D 算法, 所需的改变仅是将 (P2) 中 ER 价格 $b_i (i = 1, 2, \dots, m)$ 设为很大的值即可. 对于 H^T 算法, 用 p_i 代替 b_i 就足够了. 另外, 在 LS 模型中做短视捆绑决策时可以借鉴 ER 模型中的分解性质确定最优价格. 也就是说, 最优捆绑包是根据 $\arg \max_j (p_{ij} - p_i) \bar{F}_{ij}(p_{ij})$ 来选择, ij 捆绑包的最优价格是通过对文献 [10] 中最优化问题 (9)-(10) 的求解获得, 将这样一种近似分解的启发性算法表示为 H^S .

3.2.2 算法的有效性比较及算例分析

接下来关注应用于 LS 模型的 $H^M, H^D, H^T, H^S, H^{DRM}$ 和 H^{DRO} 算法的性能数值算例分析结果. 在算例分析中所用的数据与上面 ER 模型算例分析曾使用过的一套数据基本相同, 只是去掉了 ER 模型中的紧急补货成本 b_i . 算法的相对平均性能差异 $\varepsilon_M, \varepsilon_D, \varepsilon_T, \varepsilon_S, \varepsilon_{DRM}, \varepsilon_{DRO}$ 分别为 9.25%, 4.93%, 0.27%, 2.18%, 7.95% 和 0.26%. 表 8 和表 9 说明了同 ER 模型中实例一样, 两阶段算法和 DRO 算法的性能最为突出, 在最差的实例中的相对性能差异也仅为 1.8% 和 1.55%. 对比表 3 和表 8, 也观察到确定型启发性算法的性能在 LS 模型中更差一些, 分解启发性算法性能排到了第三的位置.

表 8 LS 模型下算法关于存量有效性的平均性能差异 (%)

γ	-0.8	-0.3	0	0.3	0.8
ε_T	0.03	0.20	0.43	0.44	0.24
ε_S	0.86	1.52	3.01	2.94	2.56
ε_D	2.93	4.75	8.75	5.62	2.59
ε_M	10.64	10.26	10.16	8.60	6.58
ε_{DRM}	12.70	10.78	8.42	5.42	2.43
ε_{DRO}	0.01	0.01	0.13	0.63	0.50

表 9 LS 模型下算法关于到达率 λ_2 的平均性能差异 (%)

λ_2	0.1	0.225	0.35	0.6
ε_T	0.33	0.11	0.13	0.38
ε_S	2.53	1.57	1.64	2.08
ε_D	5.04	4.89	4.52	5.20
ε_M	9.22	9.35	9.43	8.93
ε_{DRM}	7.99	7.90	7.88	7.96
ε_{DRO}	0.29	0.05	0.30	0.25

4 结论

在标价销售渠道中, 考虑顾客购买产品时的两种选择: 以标价 p_i 购买产品 i , 或者以标价 p_{ij} 购买包含该产品的捆绑包. 从服务产品卖方角度, 将销售周期分为若干决策单元, 运用动态规划和组合优化理论, 针对是否补充缺货分别建立了紧急补货 (ER) 模型和失销 (LS) 模型. 通过三个产品数量的例子证明了动态捆绑问题的复杂性, 提出采用启发式求解算法的方式解决动态定价和捆绑的决策问题. 通过算例比较了捆绑和定价算法的有效性和稳健性, 结果表明在产品数量不多时两阶段算法是最佳选择, 在产品数量较多或者顾客对产品捆绑价格

敏感性较低时损耗率算法中的 DRO 算法更有效. 算法分析有助于选择合适的算法对捆绑包结构和捆绑包价格进行计算. 进一步研究可考虑针对服务产品销售企业面临的不同情境选择合适的算法进行实证分析.

参考文献

- [1] Jeffery I. McGill, Garrett J. van Ryzin. Revenue management: research overview and prospects [J]. *Transportation Science*, 1999, 33(2): 233-256.
- [2] Bitran G, Caldentey R. An overview of pricing models for revenue management [J]. *Manufacturing & Service Operations Management*, 2003, 5(3): 203-229.
- [3] Kalyan Talluri, Garrett J. van Ryzin. Revenue management under a general discrete choice model of consumer behavior [J]. *Management Science*, 2004, 50(1): 15-33.
- [4] Dan Zhang, William L. Cooper. Revenue management for parallel flights with customer-choice behavior [J]. *Operational Research*, 2005, 53(3): 415-431.
- [5] 林志柄, 张岐山. 零售商的动态定价和服务模型分析 [J]. *中国管理科学*, 2011, 19(6): 73-78.
- [6] 刘迪, 杨华龙, 张燕. 多节点集装箱海铁联运动态定价决策 [J]. *系统工程理论与实践*, 2014, 34(1): 104-114.
- [7] Huiling Chung, Yanshu Lin, Jinli Hu. Bundling strategy and product differentiation [J]. *Journal of Economics*, 2013, 108(3): 207-229.
- [8] Abraham Hollander, Thierno Diallo. Pricing and bundling of shared information goods: the case of cable channels [J]. *Review of Industrial Organization*, 2013, 42(1): 25-43.
- [9] Michael Benisch, Tuomas Sandholm. A framework for automated bundling and pricing using purchase data [J]. *Social Informatics and Telecommunications Engineering*, 2012, 80: 40-52.
- [10] 杨清清, 李金林, 卢涛. 基于在线捆绑的易逝品动态定价模型研究 [J]. *经济数学*, 2012, 27(4): 1-7.

Heuristic Approaches to Dynamic Pricing Model for Perishable Products Based on Online Bundling

YANG Qing-qing, OU Chao-min, GUO Teng-da

(College of Information System and Management, National University of Defense Technology, Changsha 410073, China)

Abstract: We illustrate the complexity of the dynamic packaging problem for ER and LS model through three-product case. Propose heuristic approaches to solve dynamic packaging and pricing problem. We use an extensive numerical study to test the effectiveness of packaging and pricing / packaging heuristics for ER and LS model, which is myopic packaging and pricing heuristic, static deterministic approximation, a two-stage heuristic and the depletion ratio heuristic. We denote by the two-stage approach is best when the products are limited in number. On the other hand, the DRO heuristic may be a much more effective policy in cases in which the number of products is high or the estimated customer price sensitivity for product packages is low. It is helpful for select an appropriate approach to solve the combinatorial optimization problem.

Keywords: dynamic pricing; online bundling; servicing products; heuristics analysis