

汽车零部件 Milk-run 车辆调度优化模型和算法

王旭¹, 陈栋², 王振锋²

(1. 重庆大学 贸易与行政学院, 重庆 400030; 2. 重庆大学 机械工程学院, 重庆 400030)

(chendong222@126.com)

摘要:为了寻求汽车零部件采用循环取货时车辆的最优路径,提出将每个供应商零部件循环、分批提取使车辆尽可能满载的建模思路,建立具有车辆容积、车辆到达时间窗、供应商供货动态时间窗、车辆最大行程约束的车辆调度优化模型,设计了用于求解该模型的改进启发式节约算法。最后通过算例验证了多重约束模型及算法的有效性。

关键词:循环取货;动态时间窗;满载提取;启发式节约算法

中图分类号: TP301.6 **文献标志码:** A

Scheduling optimization model and algorithm for Milk-run auto parts

WANG Xu¹, CHEN Dong², WANG Zhen-feng²

(1. College of Trade and Administration, Chongqing University, Chongqing 400030, China;

2. College of Mechanical Engineering, Chongqing University, Chongqing 400030, China)

Abstract: To seek the optimal path for the vehicles to take delivery of auto parts under the Milk-run, a modeling idea that each components supplier's spare parts were delivered by the way of circular and batch delivery to make as full use of the vehicle as possible was put forward. The optimizing model of vehicle routing problem was established with the constraints of vehicle cubage, arriving time window, supplier supplying dynamic time window and maximum running distance. After that, a heuristic saving algorithm (or C-W algorithm) was designed to provide a solution to the model. Finally, one example was given to prove the validity of the model and algorithm.

Key words: Milk-run; dynamic time window; full-taking delivery; heuristic Clark-Wright (C-W) algorithm

0 引言

目前,国内越来越多的汽车制造企业开始专注于企业上游零部件供应物流整合管理服务的第三方物流集配中心(supply-hub)^[1],对于物料需求计划(Material Requirement Planning, MRP)中的需求零部件采取与第三方物流企业合作共同引进 Milk-run, Milk-run 又叫循环取货、牛奶式取货、集货配送等,是制造企业为了降低整条供应链的物流成本、提高车辆装载率和物料供应的敏捷性而采用的一种优化的进料物流模式。这种提货模式先按照最优的运输方案到选择好的供应商处提取原料或零部件,并将其集中到区域分拨中心(Regional Distribution Center, RDC),然后再进行拼装配送到主机厂。这样不仅可以实现零部件的准时化供应,库存最低,更重要的是降低了由供应商自己供货带来的较高社会物流成本。

国外对于 Milk-run 在汽车制造业上游供应链中已经取得广泛的应用,像丰田、福特、通用等企业将 Milk-run 引入到入厂物流环节,实现了丰田的准时生产(Just In Time, JIT),压缩了福特的存货以及送货时间,节约了通用等企业的入厂物流成本^[2]。限于 Milk-run 自身特点,以及国内供应物流发展水平等客观条件的影响, Milk-run 这种优化的提货模式,在国内的应用范围仍然比较窄,仅以长安福特与长安民生物流合作,上海通用与安吉天地(Anji-TNT)合作为代表的几家合作

企业在 Milk-run 运作上面逐步取得了明显效益。

国外在学术方面研究较早,但不是专门针对汽车制造业^[3-4]。其中文献[4]研究了基于循环取货模式的实时车辆配送系统的参数设置,并通过实验设计确定参数,在不同的 Milk-run 情景中验证实时车辆配送系统,并通过完全的 ANOVA (ANalysis Of VAriance) 分析检验了各参数的设置。最终确定当初初始配送模块采用最佳适应算法、内部路线改善模块采用 2-Exchange 算法时,系统的效果最佳。国内的研究中,文献[5]通过引入线边最大库存的限制,并最小化运输、库存成本,建立了国内汽车零部件入厂物流中循环取货路径规划的数学模型并进行求解;文献[6]将某发动机厂零件循环取货系统设计中的路线分配、取货车辆分派和时间安排问题,归结为有车辆容量限制的车辆路径问题(Vehicle Routing Problem under Capacity, CVRP)的优化,建立模型并进行求解;文献[7]研究了循环取货方式下多供应商、集配中心、制造商组成的三级供应链的成本优化问题。

国内外关于 Milk-run 的研究中,还没有将 JIT 供应和成本控制最大地平衡起来,现实运作过程中也出现很多顾此失彼的情况。Milk-run 研究中,很少有文献涉及使单位车辆尽可能满载的同时限制车辆的最大行程时间,并在此基础上追求成本最低的情况;有些文献开始研究允许单位供应商处的货物分批次地由两辆车进行提取,使车辆尽可能满载的情况。文献[8]将车辆载重作为约束,建立了将单位供应商处的货

收稿日期:2010-09-07;修回日期:2010-11-01。

基金项目:国家 863 计划项目(2006AA04A123);教育部人文社科青年基金资助项目(09YJC630246)。

作者简介:王旭(1963-),女,四川南充人,教授,博士,主要研究方向:物流基本理论、企业项目管理;陈栋(1984-),男,河南商丘人,硕士研究生,主要研究方向:交通运输物流网络工程;王振锋(1982-),男,河南南阳人,博士研究生,主要研究方向:物流作业流程管理。

物分批由两辆车提取的模型,并在原始节约算法基础上改进了算法,并通过实例验证了该种算法比供应商只允许一辆车提取时的情况下更加优化。但是仍然缺少车辆和供应商双重时间窗以及单位车辆的单程时间约束等约束。

调研发现,国内汽车制造企业在运作 Milk-run 提货过程中经常出现以下问题:车辆相对装载不高;某条线路的车辆并未满载,但其连接任何其他供应商都会超容,线路安排效率低;车辆过早或过晚回到区域配送中心(Regional Distribution Center, RDC);配送成本仍然很高;供应商不能按时间窗供货等。这些严重影响了国内汽车制造企业和物流企业应用 Milk-run 降低物流成本和提高物流供应准时性的效果。

本文主要根据国内汽车制造企业物流模式的特点,针对运作 Milk-run 时仍然出现的这些问题,在原有研究文献的基础上,以最小成本为目标函数,在车辆到达时间窗的基础上增加供应商供货动态时间窗和每辆车循环提货的总时间约束,建立单位供应商货物被分批提货、具有动态供货时间窗、实现最低成本时更多地平衡 JIT 供货需求的模型,并在原有启发式算法的基础上,通过定理 1,设计出该模型的改进算法,最终通过实例对比分析,验证了算法更加优化的结论。

1 问题描述

为了更好地针对国内汽车行业运作 Milk-run 中出现的这个问题建立模型,将模型问题描述为:物流公司 C 负责将汽车制造公司 F 的需求零配件用 m 辆车,从 n 个供应商 $p_i (i = 1, 2, \dots, n)$ 处通过 Milk-run 提取到 C 公司 RDC p_0 处,然后在 RDC 将零配件分类、再包装并按需求时间配送到 F 公司主机厂,第 i 个供应商处的供应量为 Q_i 。对车辆 k 到达的时间窗约束为:允许车辆最早到达时间为 e_{ki} ,最迟到达的时间为 δ_{ki} ;当第 i 个供应商有多个车辆提货时,产生的供货时间窗为:允许 i 的最早供货时间为 T_{ei} ,最迟供货时间为 T_{li} ;每辆车从出发到回到 RDC 最长时间限制为 t_k ,供应商 i 和 j 间运输费用为 c_{ij} ,距离为 d_{ij} ,时间为 t_{ij} ,每辆车的额定容积为 $S_k (k = 1, 2, \dots, m)$;要求每辆在尽可能满载、实现上述时间约束条件下寻求最优提货路线。

2 模型建立

2.1 假设条件

根据算法的特点,建模之前,做以下特殊说明:

1) 主机厂占据主动权,订单下后,三方协调时间以满足主机厂的准时需求为主;

2) 提货过程中的零部件种类可不同,但由同一规则包装;

3) 考虑汽车零部件的特点,一般满载情况下,零部件的载重量不会超过汽车载重限制,因此在约束中,插入容积约束替代载重约束;

4) 假设运输费用与距离成线性关系,即 $c_{ij} = ad_{ij} + b$,其中 a, b 为相关参数;

5) 每辆车在行驶过程中平均速度相同,超过其最大行程时间(最大行程距离)限制后,整体提货效率降低。

2.2 约束描述

建立模型时,为了实现该模型的目的,必须有下列约束限

制:车辆到达的时间窗限制、供应商的供货动态时间窗限制、车辆容积限制和单程时间限制。

2.2.1 时间约束描述

车辆 k 经由供应商 i 到供应商 j ,设由于经 i 到达 j 的推迟量或提前量为 e_{kj} ,则 $e_{kj} = h_{ki} + t_i + t_{ij} - h_{kj}$ 。其中: h_{ki} 表示车辆 k 到达 i 的时间; t_i 为车辆在 i 处的提货时间。 $e_{kj} < 0$ 表示车辆提前到达; $e_{kj} > 0$ 表示车辆延迟到达; $e_{kj} = 0$ 表示车辆准时到达。令 j 后的供应商为 r , δ_j^- 为在 r 处不需要等待,车辆 k 到达 j 的最大允许提前量,则 $\delta_j^- = \min_{r \geq j} \{h_{kr} - e_{tr}\}$; δ_j^+ 为车辆 k 到达 r 不延迟时,到达 j 的最大允许推迟量,则 $\delta_j^+ = \min_{r \geq j} \{l_{tr} - h_{kr}\}$ 。若 $e_{kj} < 0$,则当 $|e_{kj}| \leq \delta_j^-$ 时,车辆在 r 处不需要等待,当 $|e_{kj}| > \delta_j^-$ 时,车辆在 r 处需要等待;若 $e_{kj} > 0$,则当 $e_{kj} \leq \delta_j^+$ 时,车辆到达 r 处不延迟,当 $e_{kj} > \delta_j^+$ 时,车辆延迟到达 r 处,这些即为车辆到达的静态时间窗^[9]。

本文考虑为了更大限度地满足车辆满载,将供应商处的零部件拆分提取。因此存在多辆车到达同一个供应商处提货的情况,由于供应商的装载能力限制,就必须有车辆等待其他车辆完成装载之后才能开始。随着车辆的循环,等待时间不断出现和叠加,而这个等待时间不同于允许车辆到达时间,它产生于提货时间,不妨用供应商的供货时间窗来表达该现象引起的时间要求, T_{ei} 是允许 i 最早供货时间, T_{li} 是最迟供货时间,在模型开始求解之前,这个时间窗并不确定,是动态的。实际执行过程中, T_{ei} 和 e_{li} 同时产生和消失,因此可以认为, T_{ei} 的产生取决于 e_{li} 。 T_{li} 的产生是由于供应商被拆分提货后产生的,它与允许第一辆车到达的 l_{li} 并无直接关系,当供应商只有一辆车提货时, T_{ei} 和 T_{li} 不存在, T_{li} 反映的是当出现车辆排队提取某一个供应商货物时,所有车辆最多的等待时间和供应商最多容忍的供货持续时间。因此可以认为存在以下关系 $T_{ei} = e_{li} \leq l_{li} \leq T_{li}$ 。出于实际操作,设在供应商 i 处有 y 辆车提货, t_x 为第 x 辆车在 i 处的持续时间(包括等待和提取时间),则 $l_{li} + \sum_{x=1}^{y-1} t_x = T_{li}$ 。当 $x = 1$ 时, $l_{li} = T_{li}$,即不产生 T_{li} ;当 $x > 1$ 时,令 i 处的零部件连续全部提取所需要的时间为 t_i ,则

$\sum_{x=1}^y t_x - t_i$ 为车辆在 i 处产生的等待时间。该等待时间可能导致某车辆为了在 i 处提货,使得其提取的供应商集合为非最优。

令所有车辆在 i 处等待时间上限为 Δ_i ,于是当 $\sum_{x=1}^y t_x - t_i \leq \Delta_i$

时,将 $l_{li} + \sum_{x=1}^{y-1} t_x$ 赋予 T_{li} ;当 $\sum_{x=1}^y t_x - t_i > \Delta_i$,令 $x - 1 = x$,如此循环直到 $\sum_{x=1}^y t_x - t_i \leq \Delta_i$ 。

为了更多地平衡 JIT 供应,除了上述两组时间窗约束,还应限制每辆车从出发到回到 RDC 的最长时间 t_k ,根据假设 2.1 节中的条件 5),将最大行驶时间约束表现为车辆的最大行程约束。

2.2.2 车辆最大行程约束描述

考虑到提货车辆在供应商处停留时间过长可导致整个供应链延迟,无法满足主机厂 JIT 需要,并且行驶时间过长可导致司机疲劳驾驶、机器受损等情况,影响提货整体效率,基于

上述情况,对车辆的最大行程进行约束限制, $\sum_{f=1}^{n_k} d_{k_{f-1}k_f} + d_{k_{n_k}0} \leq R_k$, 且 $n_k \neq 0$, 其中 $d_{k_{f-1}k_f}$ 表示由车辆 k 提货的供应商中任两个连续供应商或 RDC 之间的距离, $d_{k_{n_k}0}$ 表示由车辆 k 提货的最后一个供应商与 RDC 间的距离。

2.2.3 容积约束描述

设每个零部件占用空间体积为 v (每个零部件都由规则统一包装), 供应商 i 处的零部件数量为 $Q_i (i = 1, 2, \dots, n)$, 每辆车的额定容积为 S_k 。为根据车辆容积限制安排线路, 需要预先对车辆数估计, 设车辆数为 m , 由公式 $m = \text{int} \left[\sum Q_i v / \alpha S_k \right] + 1$ 对车辆数进行确定, $\text{int}[\]$ 表示不大于中括号内数字的最大整数, α 为提货装车的复杂度及约束多少的估计, 通常 α 取 $[0.8, 1]$ 。

令 n_k 表示车辆 k 提货的供应商数, 当 $n_k = 0$ 时, 表示车辆 k 没有参加提货。令 R_k 表示车辆 k 提货的供应商集合, 当 $n_k = 0$ 时, $R_k = \emptyset$; 否则, $R_k = \{k_1, k_2, \dots, k_{n_k}\} \subseteq \{1, 2, \dots, n\}$, 其中 $k_f (f = 1, 2, \dots, n_k)$ 表示该供应商在 k 的提货线路中顺序为 f , 那么 Q_{k_f} 则表示 k 所要提货的第 k_f 个供应商处的零部件数量。为了实现车辆尽可能满载, 需满足:

$$\sum_{f=1}^{n_k} v Q_{k_f} \geq V_k, \left(\sum_{f=1}^{n_k} v Q_{k_f} - V_k \right) + \sum_{f=1}^{n_{kp}} \sum_{k_p=1}^m v Q_{k_f} \leq m V_k;$$

$$m = m - 1, n_k, n_{kp} \neq 0, kp \neq k$$

即车辆满载之后, 供应商剩余零部件与其他零部件之和不大于剩余车辆的额定容积之和。

2.3 模型建立

基于以上约束描述, 建模之前首先定义关于 n_k 的二进制函数。sign 为符号函数, 功能为根据参数的正、负、零的属性, 返回结果分别为 1、-1、0, 由于 $n_k \geq 0$, 所以 $\text{sign}(n_k) = 1 (n_k \geq 1)$ 或 $0 (n_k = 0)$ 。

在提货时, 一个供应商允许多辆车进行提货, 此时在所有车辆提货集合中, 任意两个集合至多只有一对相同元素同时为一个供应商^[10], 见定理 1。

定理 1 在最优提货组合集 R 中, 其中 $R = \{R_1, R_2, \dots, R_k, \dots, R_m\} (k = 1, 2, \dots, m)$, 任意两不同的车辆提货集合 $R_{kp} = \{k_{p1}, k_{p2}, \dots, k_{pn_{kp}}\}$ 和 $R_{kq} = \{k_{q1}, k_{q2}, \dots, k_{qn_{kq}}\}$ 的元素中至多只有一对元素指向同一个供应商, 其中 $kp = 1, 2, \dots, m$, $kq = 1, 2, \dots, m$, 且 $kp \neq kq$ 。

证明 假设 R_{kp} 和 R_{kq} 两个车辆最优提货供应商集合中, 有两对元素分别指向两个不同的供应商, 即 R_{kp} 和 R_{kq} 中有两个供应商是相同的。为不失一般性, 令 $k_{p\alpha}$ 与 $k_{q\beta}$ 指同一供应商 i , $k_{p\alpha}$ 和 $k_{q\beta}$ 同指供应商 j , 令车辆 kp 在 i 处和 j 处提货量分别为 up_i, uq_j , 车辆 kq 在 i 处和 j 处提货量分别为 uq_i, uq_j , 此时不妨设 $up_i = \min \{up_i, up_j, uq_i, uq_j\}$, 那么可以调整 R_{kp} 中不包含 $k_{p\alpha}$, 其他车辆提货供应商集合不变, 调整后车辆 kp 线路集 R_{kp} 将变为 $\overline{R_{kp}}$, $\overline{R_{kp}}$ 不包含元素 $k_{p\alpha}$, 若保持 i 处和 j 处总提货量不变, 此时新提货安排为 $\overline{up_i} = 0, \overline{up_j} = up_i + up_j, \overline{uq_i} = uq_i + up_i, \overline{uq_j} = uq_j - up_i$, 即车辆 kp 在 $k_{p\alpha}$ 处增加提货量 up_i , 车辆 kq 在 $k_{q\beta}$ 处增加提货量 up_i , 在 $k_{q\beta}$ 处减少提货量 up_i , 此时便调整为了新的最优提货线路集 R^* , 其中 $R^* = \{R_1, \dots, \overline{R_{kp}}, \dots, R_{kq}, \dots, R_m\}$, 由于 $R_{kp} \leq \overline{R_{kp}}$, 所以 $R \leq R^*$, 根据前述假设的提货距离与成本成正比, 则新的 R^* 下的成本 C^* 小于 R 下的成本 C ,

即 $C^* \leq C$, 这与 R 为最优车辆提货供应商集合相矛盾, 即 R_{kp} 和 R_{kq} 元素中至多只有一对元素指向同一个供应商。

基于上述定理, 建立模型如下:

$$\min Z = \sum_{k=1}^m \left[a \left(\sum_{i=1}^{n_k} d_{k_{i-1}k_i} + d_{k_{n_k}0} \right) + b \right] \cdot \text{sign}(n_k) \quad (1)$$

$$\text{s. t. } \sum_{f=1}^{n_k} v Q_{k_f} \geq V_k, \left(\sum_{f=1}^{n_k} v Q_{k_f} - V_k \right) + \sum_{f=1}^{n_{kp}} \sum_{k_p=1}^m v Q_{k_f} \leq m V_k;$$

$$m = m - 1, n_k, n_{kp} \neq 0, kp \neq k \quad (2)$$

$$\sum_{f=1}^{n_k} d_{k_{f-1}k_f} + d_{k_{n_k}0} \leq M_k; n_k \neq 0 \quad (3)$$

$$\bigcup_{k=1}^m R_k = \{1, 2, \dots, n\}, \sum_{k=1}^m n_k = n; 0 \leq n_k \leq n \quad (4)$$

$$R_{kp} \cap R_{kq} \leq 1; kp, kq \in [1, m], kp \neq kq \quad (5)$$

$$e_{ii} \leq h_{ii} \leq l_{ii}, l_{ii} + \sum_{x=1}^{y-1} t_x = T_{ii}, \sum_{x=1}^y t_x - t_i \leq \Delta_i \quad (6)$$

$$\text{sign}(n_k) = \begin{cases} 1, & n_k \geq 1 \\ 0, & n_k = 0 \end{cases} \quad (7)$$

其中: $\min Z$ 为模型的目标函数, 即在完成全部提货时费用最小; 式(2)为车辆的容积约束, 式(3)为车辆的最大行程约束, 式(4)为每个供应商的提货需求必须满足, 式(5)表示任意两个提货最优线路最多只有一个交点, 式(6)为供应商供货动态时间窗, 式(7)为 n_k 的符号函数。

3 算法设计与分析

3.1 算法设计

根据模型的特点, 该问题属于 Milk-run 车辆调度问题 (Vehicle Routing Problem, VRP), Milk-run VRP 的算法可以从配送领域的 VRP 算法中寻找, 一般 VRP 算法分为精确算法和启发式算法两大类。VRP 属于强 NP 问题, 运用精确算法求解计算量会随着问题规模的增大而呈指数增加, 很难找到目标最优解。而启发式算法强调满意解, 不刻意追求最优性, 因此可以在有限时间里, 找到满意的次优解或可行解^[11]。而本文模型正是在使车辆尽可能满载和 JIT 供应条件的成本最低情况, 因此实现模型的精确结果十分困难, 所以本文选择启发式算法来解决模型。

启发式算法的种类很多, 其中节约算法是一种适用于解决车辆路线安排的启发式算法, 该算法是由 Clark 和 Wright 在 1964 年提出, 因此又叫 C-W 算法。其原理从 Milk-run 提货模式方面理解: 车辆 k 从 p_0 去 p_i 处提货, 令 s_{ij} 为连接这两个供应商合并提货的运输距离比车辆分别往返这两个供应商的运输距离的节约值, 即 $s_{ij} = d_{0i} + d_{0j} - d_{ij}$, 这就是节约量公式, 且 $s_{ij} \geq 0$ 。根据模型, 设计改进的启发式节约算法流程如下。

步骤 1 将 RDC 与各个供应商相连, 构成单程往返的初始路线。

步骤 2 计算连接任意两个供应商 i, j 后的节约值 $V(i, j)$, 并令集合 $S = \{V(i, j) | V(i, j) \geq 0\}$ 。

步骤 3 判断是否 $S = \emptyset$, 若是则终止, 否则选择节约量最大的两个供应商。

步骤 4 计算连接 i, j 后车辆 k 的总行程 S_k , 并比较是否 $S_k \leq M_k$, 若是转下一步, 否则转步骤 15。

步骤 5 判断通过供应商 j 的车辆 k 是否是第一辆到达该

供应商,若是转下一步,否则转步骤 7。

步骤 6 计算 e_{kj} , 若 $e_{kj} = 0$, 转步骤 8; 若 $e_{kj} < 0$, 当 $|e_{kj}| \leq \delta_j^-$ 时转步骤 8, 否则转步骤 14; 若 $e_{kj} > 0$, 当 $e_{kj} \geq \delta_j^+$ 时转步骤 8, 否则转步骤 14。

步骤 7 计算车辆 k 到达供应商 j 的时次 x , 并比较 $\sum_{x=1}^y t_x - t_i$ 和 Δ_i , 若 $\sum_{x=1}^y t_x - t_i \leq \Delta_i$, 将 $l_i + \sum_{x=1}^{y-1} t_x = T_k$, 转步骤 8, 否则转步骤 15。

步骤 8 计算连接 i, j 后车辆 k 提货的供应商集合 R_k 中零部件总量 Q_k 所占空间体积 $Q_k \times v$, 并计算是否 $Q_k \times v \geq V_k$, 若是转下一步, 否则转步骤 12。

步骤 9 令 $m - 1 = m$, 并计算 $\left(\sum_{j=1}^{n_k} vQ_{kj} - V_k \right) + \sum_{j=1}^{n_{kp}} \sum_{k=1}^m vQ_{kj} \leq mV_k$ 是否成立, 若成立转下一步, 否则转步骤 15。

步骤 10 连接 i, j 两个供应商, 并使车辆 k 满载, 完成集合 R_k , 转下一步。

步骤 11 将被拆分的供应商 j 剩余零部件作为新的供应商并入剩余供应商集合中, 并使 $S = S - V(i, j)$, 转步骤 2。

步骤 12 连接 i, j 两个供应商, 若 i, j 分别为线路中的始点和终点, 转下一步; 若 i 不是始点, j 为终点, 转步骤 14。

步骤 13 选择与这条线路上首尾节点节约量最大的供应商 r , 连接到该新线路中, 并将 i 赋予 j , 将 j 赋予 r , 转步骤 4。

步骤 14 选择与供应商 j 节约量最大的供应商 s , 将 i 赋予 j , 将 j 赋予 s , 转步骤 4。

步骤 15 令 $S = S - V(i, j)$, 转步骤 3。

3.2 算法分析

该算法是按照模型目标进行的设计, 这种改进的节约算法与传统的节约算法不同之处在于, 供货商处的货物允许多辆车进行提取, 并且尽可能使得车辆满载, 同时保证车辆最长行程时间满足约束。当算法完成一辆车的循环后, 有剩余货物的供应商将会被重新并入剩余供应商集合中, 再次形成一个新的提货线路安排问题。

这种情况下可能使得车辆为了满足单次时间约束而放弃满载。这就避免了有些算法中, 为了实现满载而导致整个供应链延迟的情况。另外模型中的特殊约束条件以及将供应商处的货物分批使得算法变得复杂, 由于步骤和关联太多, 可能会因为数据的不完整使得整个计算过程出现差错, 导致结果的不准确性。算法的再改进和再精简还需要进一步的研究。

4 算例分析

本文以物流企业 C 为汽车制造企业 F 引进 Milk-run 提货项目为背景, 对该项目在某地区提货数据进行了统计分析, 综合评定出 10 家大的供应商。考虑到单位车辆提货时间约束, 限制每辆车一次往返最多 5 h, 按照平均 60 km/h 的速度, 限制车辆一次运行 280 km 以内。对零部件提货车辆一般为托卡, 若选择拖挂长度为 12 m 的托卡去提货, 零部件都由统一箱式包装, 单位箱数有效体积为 3 m³ (托盘计算在内), 因此由公式 $m = \text{int} \left[\left(\sum Q_i v \right) / (\alpha S_k) \right] + 1$ 初步估算所需车辆数 $m = 5$ (其中 α 取 0.9)。其具体模型数据如表 1。其中位置坐标为根据 RDC 与各供应商互相之间的距离, 选取随机点为原

点的相对坐标。

根据表 1 数据, 按照上述算法, 实现的模型优化结果如表 2 所示, 其提货路径指示图如图 1 所示。

表 1 模型计算相关参数

RDC 与供应商代码	位置相对坐标	主机厂对供应商需求箱数	供应商处零部件总提货时间 t_i/h	供应商处允许车辆达到时间窗 $[e_u, l_u]$	各供应商处允许所有车辆等待的时间上限 Δ_i/h
p_0	(60, 140)	—	—	6:30 从 RDC 出发	—
p_1	(35, 110)	11	0.7	[7:00, 9:00]	0.45
p_2	(45, 30)	14	0.6	[8:00, 10:00]	0.50
p_3	(50, 125)	18	0.8	[8:00, 10:00]	0.50
p_4	(108, 135)	8	0.4	[6:30, 10:30]	0.40
p_5	(130, 88)	12	0.6	[8:00, 14:30]	0.60
p_6	(90, 115)	16	0.9	[7:00, 12:30]	0.80
p_7	(95, 66)	10	0.5	[7:30, 11:30]	0.60
p_8	(80, 165)	6	0.3	[7:00, 10:00]	0.50
p_9	(55, 175)	12	0.6	[7:00, 10:30]	0.50
p_{10}	(30, 190)	10	0.5	[7:00, 11:30]	0.65

表 2 对供应商零部件分批提取时的路径优化结果

路径/车辆编号	供应商编号及时间安排	线路全长/km	载容量/m ³	装载率/%	装车数 n_k
线路 1 车辆 1	$p_0-p_2-p_7-p_4-p_0$ 6:30-8:43-10:33-11:52-12:55	261	90	100	3
线路 2 车辆 2	$p_0-p_4-p_5-p_6-p_0$ 6:30-7:18-8:34-9:58-11:07	187	90	100	3
线路 3 车辆 3	$p_0-p_{10}-p_9-p_8-p_0$ 6:30-7:05-8:10-9:36-10:26	146	84	93	3
线路 4 车辆 4	$p_0-p_3-p_1-p_0$ 6:30-6:48-7:57-9:18	78	87	97	2

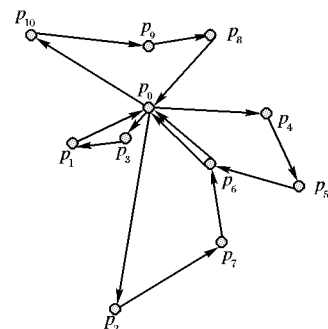


图 1 允许拆分提货时优化路径结果

此时提货车辆为 4 辆, $n_5 = 0$, $R_5 = \emptyset$, 总装载率为 97.5%, 提货线路总长 $S = 681$ km, 节约距离 $S_a = 1098 - 681 = 417$ km。运行过程中, 当连接 p_2 和 p_7 后构成线路 $p_0-p_2-p_7$, 由于车辆 1 未达到满载, 需要寻找与 p_2 和 p_7 节约值最大的供应商, 此时 $V(p_7, p_5) = 128$ km 最大, 但是线路 $p_0-p_2-p_7-p_5-p_0$ 总路程 301 km 超过了最大里程约束, 循环两次, $p_0-p_2-p_7-p_6-p_0$ 满足各约束条件, 此时 $R_{k1} \cap R_{k2} = 1$, 提货供应商集合 R_{k1} 和 R_{k2} 中的 $k1_3$ 和 $k2_3$ 同时指向 p_6 ; 当行程线路时 $p_0-p_{10}-p_9-p_8-p_0$ 车辆 3 并未满载, 现在只能寻找 p_3 和 p_1 使车辆 3 满载, p_8 为此时线路的终点, 只能寻找与 p_8 节约值最大的供应商连接, 当连接供应商 p_3 后, 线路违反 p_3 第一次到达时间窗, 而 $V(p_8, p_1) = 0$, 此时算法选择了以非满载来满足时间窗和节约值最大的条件。

(下转第 1132 页)

写入 RAM 中。图中共传输了 6 个数据,可再用主控读方式将数据读出,验证与写入数据传输是否一致。

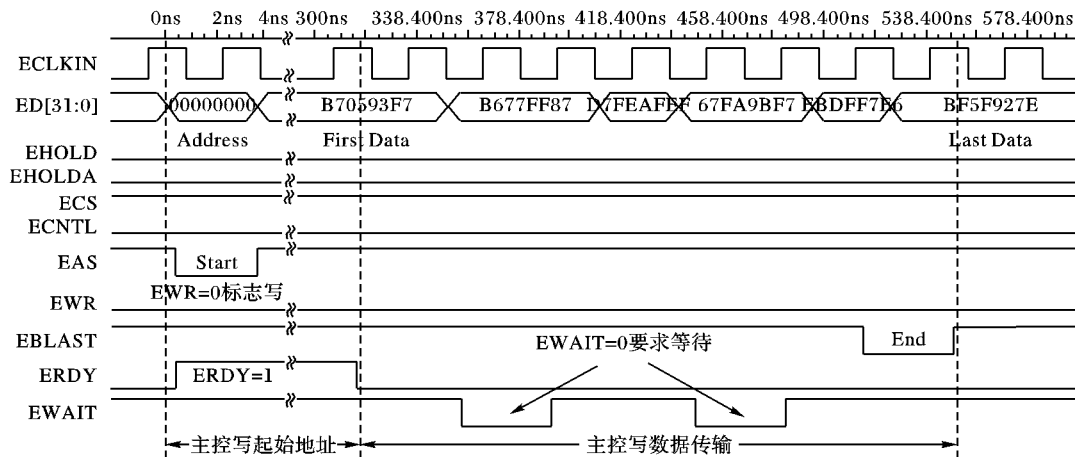


图6 YHFT-DSP 通过 E-Bus 同步主控批量写访问波形

4 结语

本文为自主研制的 YHFT-DSP 芯片设计了一套 E-Bus 接口仿真平台,基于 FPGA 实现了 E-Bus 与 USB 接口之间数据传输的协议转换和控制,经过 RTL 级功能验证与软硬件协同系统调试,保证了硬件逻辑设计的正确性与仿真平台的正常运行。目前该仿真平台能正确、稳定地工作,提供了操控界面友好的仿测真试环境,满足了 YHFT-DSP 芯片投片后对其 E-Bus 接口测试的需求,也为基于 E-Bus 的应用开发搭建了较好的平台。

参考文献:

[1] 陈书明,李振涛,万江华,等.“银河龙腾”高性能数字信号处理器

研究进展[J]. 计算机研究与发展, 2006, 43(6): 993 - 1000.

- [2] 沈戈,高德远,樊晓桢. 数字信号处理器(DSP)结构设计及发展趋势[J]. 计算机工程与应用, 2003, 39(7): 4 - 7.
- [3] (美) Texas Instruments Incorporated. TMS320VC55x 系列 DSP 的 CPU 与外设[M]. 彭启琮,武乐琴,张舰,译. 北京:清华大学出版社, 2005.
- [4] 黄虹. YHFT-DSP E-Bus 软硬件协同仿真平台的设计与实现[D]. 长沙:国防科学技术大学, 2010.
- [5] Xilinx. Platform Flash PROM User Guide[EB/OL]. (2008-01-15) [2010-05-25]. http://www.china.xilinx.com/support/documentation/user_guides/ug161.pdf.
- [6] 萧世文,宋延清. USB2.0 硬件设计[M]. 北京:清华大学出版社, 2006.

(上接第 1128 页)

为了验证模型的实用性以及算法的优势,本文对单车配送的含有时间窗的节约算法进行设计,并根据上述有关基础数据进行算法实现,得到的结果与分批提取的改进算法相比,改进节约算法进一步优化了结果,车辆减少一辆,装载率提高 19.5%,全程进一步节约 63 km,因为有单程时间约束,因此模型、算法更加平衡了 JIT 思想。随着供应商的数量的增加,有些路线可能不会实现满载,但是平衡了时间和节约值,使得算法更加合理。

5 结语

本文提出将每个供应商零部件循环、分批提取使车辆尽可能满载的建模思路,建立具有时间、容积和车辆最大行程时间多重约束的车辆调度优化模型,设计了用于求解该模型的改进启发式节约算法。结果表明,当允许供应商零部件拆分提货比不拆分提货,从成本、装载率、车辆数量等上面都能进一步优化,更加能够体现汽车制造业引进 Milk-run 提货模式的节约物流成本的目标,加上单程时间约束提高了运作效率,实现准时供应和生产的目标。但是算法考虑的是理想状态,并且由于模型的复杂性导致算法设计的复杂,很容易由于数据的不完整使得整个过程不准确,进一步的研究方向是算法的精简以及考虑多提货中心的情况及其处理方法。

参考文献:

[1] BARNES E, DA I J, DENG S, *et al.* On the strategy of supply hubs for cost reduction and responsiveness [R]. Atlanta, USA: Georgia Institute of Technology, 2000.

- [2] CHOPRA S, MEINDL P. Supplier chain management: Strategies, planning and operation [M]. 北京:清华大学出版社, 2001.
- [3] CHUAH K H. Optimization and simulation of just-in-time supply pickup and delivery systems [D]. Lexington: University of Kentucky, 2004.
- [4] DU T, WANG F K, LU P Y. A real time vehicle-dispatching system for consolidating milk-runs [J]. Transportation Research, Part E: Logistics and Transportation Review, 2007, 43(5): 1 - 13.
- [5] 汪金莲,蒋祖华. 汽车制造厂零部件入厂物流的循环取货路径规划[J]. 上海交通大学学报, 2009, 43(11): 1703 - 1709.
- [6] YI JUN-MIN, ZHOU JING, GAO XIAO-LIANG, *et al.* Tactical planning and optimization of a milk run system of parts pickup for an engine manufacturer [J]. Journal of Southeast University: English Edition, 2007, 23(S1): 99 - 104.
- [7] 陈建华,帅颖,林东波,等. 三级供应链循环取货成本优化模型[J]. 武汉理工大学学报: 信息与管理工程版, 2009, 31(5): 838 - 841.
- [8] 张学志,陈功玉. 车辆路线安排的改进节约算法[J]. 系统工程, 2008, 12(11): 67 - 70.
- [9] DESROCHERS M, DESROSIERS J, SOLOMON M. A new optimization algorithm for the vehicle routing problem with time windows [J]. Operations Research, 1992, 40(2): 342 - 354.
- [10] SALVESBERGH M. Local search in routing problems with time windows [J]. Annals of Operations Research, 1985, 33(4): 285 - 305.
- [11] 李军,郭耀煌. 物流配送车辆优化调度理论和方法[M]. 北京:中国物资出版社, 2001.