Public Logistics Network Protocol Design and Implementation

Ling Xiang, Michael G. Kay, and John Telford
Department of Industrial and Systems Engineering
North Carolina State University
Raleigh, NC 27695, USA

Abstract

A public logistics network (PLN) is proposed as an alternate network to private logistics networks for the ground transport of parcels. In a PLN, a pricing protocol is used to coordinate the operation of the network. The goal of the proposed protocol is to match the load that values transport the highest with the truck that can provide that service at the least cost. This protocol has been implemented using the Microsoft .NET framework. The implementation specifies DC services and the package bidding process. Total logistics cost (TLC) is used to evaluate the performance of the protocol. The protocol is compared to a FIFO protocol under several different scenarios.

Keywords
Logistics Networks, protocol design, parcel transport

1. Introduction
A public logistics network (PLN) is proposed as an alternative to private logistics networks for the ground transport of parcels. Using the analogy between the packages transported in the network and the packets transmitted through the Internet, a package would be sent from a store and then routed through a sequence of public distribution centers (DCs) located throughput a metropolitan area and then delivered to a home in a matter of hours, making a car trip to the store to get the package unnecessary. The DCs, functioning like routers in the Internet, could also be located at major highway interchanges for longer distance transport. Currently, it is common for a single logistics firm like UPS and FedEx to handle a package throughout its transport. In a PLN, the different functions of the network would be separated so that a single firm is not required for coordination. This would enable scale economies to be realized in performing each logistics function since each element of the network has access to potentially all of the network's demand [1,2].

Figure 1 shows hypothetical public logistics network with 36 public DCs covering the southeastern portion of the U.S. and connected via interstate highways. Each of the interstate DCs would serve as a hub in a sub-network of local DCs (not shown) covering the region surrounding the interstate DC [1].

2. PLN Protocols
In a PLN, each truck and package abides by a protocol that coordinates the operation of the network. The goal of the proposed protocol is to, through a package bidding process, match the load that values transport the highest with the truck that can provide that transport service at the least cost [2,4,6].

2.1 Truck Protocols
The truck protocol consists of two rules and is used to determine which truck is used to transport what load at what DC:
- Opportunity to accept or reject load is based on truck's arrival time at DC.
- After reneging, a truck can't again accept the same load until all other trucks have rejected it.

2.2 Package Protocols
The package protocol consists of three rules and is used to determine which packages are selected to join a load:
- Packages are assigned to load that maximizes resulting load bid.
- Truck's portion of a load bid does not increase after acceptance.

- Packages that rejoin load are charged previous bid amounts.

The goal for package selection is to encourage a package to submit a bid that represents its true value for transport as soon possible, thereby allowing trucks to be more responsive and discouraging multiple-bid auction-like behavior.
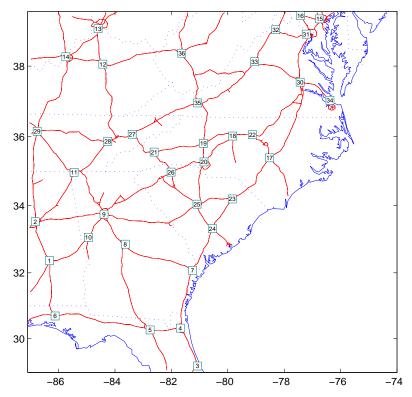

Figure 1 Hypothetical public logistics network coving southeastern portion of the USA

3. Implementation of PLN
Since Microsoft .NET framework has advanced data structures and efficient program control procedures, it is a suitable tool to implement this protocol.

3.1 Program Objective Design
Under .NET platform, every entity is implemented as an object [5]. In figure 2, each object is enclosed into a rectangle, with its properties, operations, etc. For example, *package* is an object and it has properties *StartDC* and *EndDC*, and several associated functions (e.g., *Package()*). Arcs denote the relationship between objects.

3.2 DC Configuration
The DCs play a vital role in a PLN. They provide DC services to packages and trucks that include biding information, truck offering support, truck estimated arrival time, etc.

Each DC is composed of several loading areas, which correspond to the arc between this DC and its adjacent DC. For example, in figure 3, DC 1 is adjacent to DC 2, DC 3, and DC 4. Then, within DC 1, there are three loading areas designated to packages heading to DC 2, DC 3, and DC 4. Whenever a package gets unloaded from an incoming truck, it will go to the loading area that corresponds to its next destination.
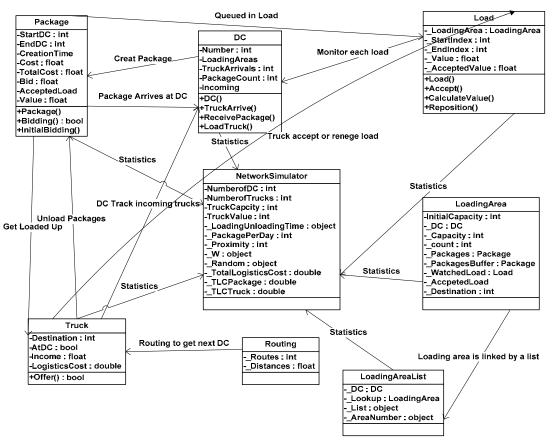
Package

- -StartDC : int
- -EndDC : int
- -CreationTime
- -Cost : float
- -TotalCost : float
- -Bid : float
- -AcceptedLoad
- -Value : float

- +Package()
- +Bidding() : bool
- +InitialBidding()

DC

- -Number : int
- -LoadingAreas
- -TruckArrivals : int
- -PackageCount : int
- -Incoming

- +DC()
- +TruckArrive()
- +ReceivePackage()
- +LoadTruck()

Load

- -_LoadingArea : LoadingArea
- -_StartIndex : int
- -_EndIndex : int
- -_Value : float
- -_AcceptedValue : float

- +Load()
- +Accept()
- +CalculateValue()
- +Reposition()

Queued in Load

Creat Package

Package Arrives at DC

Monitor each load

Truck accept or renege load

Statistics

Statistics

NetworkSimulator

- -NumberofDC : int
- -NumberofTrucks : int
- -TruckCapcity : int
- -TruckValue : int
- -_LoadingUnloadingTime : object
- -_PackagePerDay : int
- -_Proximity : int
- -_W : object
- -_Random : object
- -_TotalLogisticsCost : double
- -_TLCPackage : double
- -_TLCTruck : double

DC Track incoming trucks

Unload Packages

Get Loaded Up

Statistics

Statistics

Statistics

LoadingArea

- -InitialCapacity : int
- -_DC : DC
- -_Capacity : int
- -_count : int
- -_Packages : Package
- -_PackagesBuffer : Package
- -_WatchedLoad : Load
- -_AccpetedLoad
- -_Destination : int

Truck

- -Destination : int
- -AtDC : bool
- -Income : float
- -LogisticsCost : double

- +Offer() : bool

Routing to get next DC

Routing

- -_Routes : int
- -_Distances : float

Statistics

Loading area is linked by a list

LoadingAreaList

- -_DC : DC
- -_Lookup : LoadingArea
- -_List : object
- -_AreaNumber : object

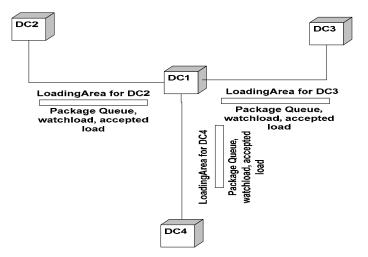Figure 2 Object design configuration

Figure 3 DC configuration

3.3 Program Flow Chart
Figure 4 reflects the trucking offering and package bidding procedure [8,10] that occurs during a period time from a truck's arrival at a DC until its departure. It involves all the protocols introduced above, and it is the core procedure in the implementation.
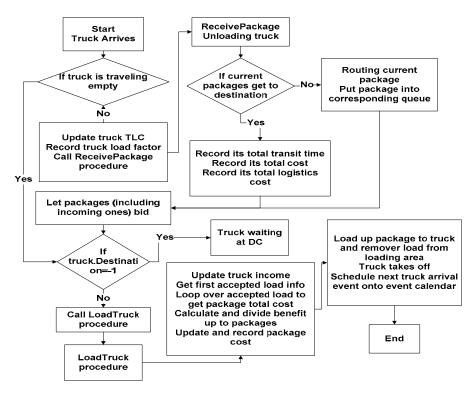

Figure 4 Program flow chart

3.4 Optimization for Model Design
Approximately 1.8 million packages are handled per day by UPS in the southeastern U.S. [11]. To compare PLN performance with real UPS network, simulation and computation at same package level are desired. Advanced data structures are incorporated to make this model run efficiently.

For example, for the package ordering procedure in each loading area, a simple array copy and array paste function is used in the beginning. However it is really inefficient and makes the program take forever to run, even with a small number of packages per day. Incorporating .NET framework's advanced data structures, such as linked lists and heaps, has allowed a more efficient algorithm to be developed.

In Figure 5, *package queue* is pointed to by linked list *A*, which originally holds packages in that loading area. Each time a package re-bids with higher value, it is moved to a heap. After the bidding process is finished, a comparison between the linked list *A* and heap takes place. A heap is an efficient data structure to order data since the maximum value is always at the top. Whichever bigger between linked list *A* and heap will be moved to linked list *B*. Then, instead of copying entire list *B* list *A*, package queue re-point to linked list *B*. In that way, linked list *A* and *B* serves alternatively as original and swap space.

4. Time Value Evaluation in PLN
How do we evaluate time value for each package stays in PLN? One natural solution is to refer to the UPS logistics network. A study for package time value in UPS logistic network has been done [3]. Suppose a package with weight of 5 lb was shipped by UPS on 12:00PM, Monday, December 18, 2006 from Raleigh, NC. It will be delivered to Tampa, FL 33601, Atlanta, GA 30305, and Charlotte, NC 28206, which represent long, medium, and short distance delivery, respectively. Comparing charges and its corresponding service provided by UPS, an average $1.22 per

hour stay in the UPS logistics network is estimated. This estimate provides a means to assign a cost to the time a package spends in-transit and, along with truck operating costs, allows network performance comparisons.
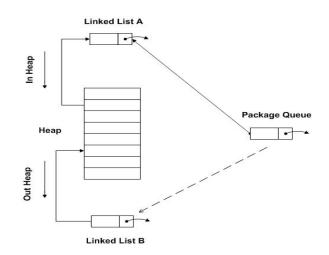

Figure 5 Efficient architecture design for package re-ordering process in loading Area

5. Comparison between FIFO and Protocol-based Control

In the existing UPS or FedEx logistics networks, transit service is distinguished by different service class categories. For simplification, in the same service class, we are assuming package is served on first come first served basis, which is also called first in and first out (FIFO). In a PLN, a protocol is implemented, which encourages and favors packages with higher bid value. Packages are competing against each other by their bid amount to get faster transit service. A comparison has been carried out between these two systems. FIFO is implemented by simply making every package bid the same amount at all time. The PLN protocol is implemented by making a package submit a bid that is proportional to its value.

Total logistics cost (TLC), which is the sum the cost associated with distance traveled by each truck and cost associated with time each package spend in the PLN, is a criterion to evaluate the performance of the PLN protocol:

$$TLC = \sum_{i=1}^{m} v_i t_i + \sum_{j=1}^{n} c_{truck} d_j \qquad (1)$$

where, m is number of packages which have been delivered by PLN, n is number of trucks, v_i and t_i are the time value for package i, c_{truck} is the truck operating cost (\$ per mile), and d_j is the distance truck j traveled.

Table 1 Comparison result between FIFO and PLN with intelligent protocol

	\$10 bid	\$5 bid	\$1 bid	1/5 of its value bid	1/10 of its value bid	1/50 of its value bid
PLN TLC (\$ million)	98.1	98.1	91.3	89.8	88.5	85.7
Pkg Delivery Mean Time(hr)	11.38	11.38	10.74	10.63	10.49	10.27
TLC Decrease				8.4%	9.8%	6.1%
Delivery Mean Time Decrease				6.6%	7.8%	4.4%

Package values are normally distributed with a mean of \$50 and a standard deviation of \$5. Table 1 shows, by implementing the protocol, TLC has decreased from \$91.3 to \$85.7 million (a 6.1% decrease), and package delivery mean time has decreased from 10.74 to 10.27 hours (a 4.4% decrease). Therefore, it is safe to conclude that PLN with protocol performs better than FIFO system in terms of total logistics cost and package delivery mean time.

6. PLN Performance under Natural Disasters

PLN performance under natural disasters, such as hurricanes, tornadoes, etc., is evaluated by comparing against a normal situation. Assume, for example, that a disaster strikes DC 5, and it happens at the fifth day and lasts one day. This causes the demand at DC 6 to increase ten times. All packages going to DC6 are bidding three times as much as in the normal situation.

Table 2 Comparison of PLN performance between normal mode and disaster mode

DC Number	Normal Mode Truck Arrivals	Disaster Mode Truck Arrivals	Comparison Number of trucks difference
0	1132	1471	339
1	703	803	100
2	385	416	31
3	1933	2185	252
4	1525	1814	289
5	698	973	275
6	2004	2014	10
7	2239	2026	–213
8	3710	3488	–222

Table 2 is only a portion (DCs 0 to 8) of the entire table that includes DCs 0 to 35. Trucks that are going through DC 5 have increased from 698 to 973, and also the DCs around DC5, such as DC4 and DC3 received many more trucks. It verifies that, in a PLN, with decentralized control, it could achieve the same performance as a centralized control network (e.g., UPS, FedEx) do in the natural disaster, or other unusual situation.

7. Conclusion and Future Work

A PLN with protocol has been successfully implemented and its performance is verified to be slightly better than FIFO by comparing TLC and delivery mean time. It achieves this using a decentralized control scheme, as compared to the centralized control used in private logistics networks. Future work will be focused on the development of intelligent package and truck agents. Agents will interact with each DC and make decisions regarding package bidding (package agents) and truck load acceptance (truck agents).

Acknowledgements

References

1. Kay, M.G. and Parlikad, A.N., 2002, "Material flow analysis of public logistics networks," Progress in Material Handling Research, 205–218.
2. Kay, M.G., 2004, "Protocol Design for a Public Logistic Network," Progress in Material Handling Research, 181–188.
3. UPS website. http://wwwapps.ups.com/calTimeCost?loc=en_US.
4. Yao, P., Durant, D., 2004, NET compact framework programming with C#, Addison-Wesley, Boston.
5. Tod Golding, 2005, Professional .NET 2.0 generics, John Wiley, Indianapolis, IN.
6. Ashish, J., 2004, Protocol design for a public logistics network, Thesis (M.S.), North Carolina State University.
7. Bansal, A., 2004, Design of a public logistics network, Thesis (M.S.), North Carolina State University.
8. Engelbrecht-Wiggans, R., et al., 1983, Auctions, bidding, and contracting: uses and theory, New York University Press, New York.
9. Ho, C., 2005, Public logistics network cost analysis, Thesis (M.S.), North Carolina State University.
10. Viscusi, W. K., 1995, Economics of regulation and antitrust, MIT Press, Cambridge, MA.
11. U.S. Census Bureau ZCTA files (http://www.census.gov/geo/ZCTA/zcta.html).