SMART CITIES:
Old Problems/New Paradigms

Classical O.R and New Paradigms:
Innovative Urban/SubUrban Mobility
Of Smart Cities
Old Problems/New Paradigms: Innovative Mobility

Summary

I. O.R Trends, New Paradigms
II. LIMOS and Innovative Mobility
III. A reference problem: Dial/Ride
IV. Standard Methods and Benchmarking
V. Extensions toward New Contexts
   V.1. Mixing Decision Levels
   V.2. Non Standard Criteria
   V.3. Non Standard Contexts
I. O.R Trends, New Paradigms

O.R -> born at the end of the 40’s, from the needs of U.S Army;
Centralized and static point of view;
Mainframes; High human computing costs;
No web, no P.C, no mobile communication devices

⇒ Linear Programming (Von Neuman/Dantzig) - Graph Theory (Berge), Complexity (Cook) + MIP (Gomory)
⇒ Polyhedral Theory (Edmonds).
Many things have changed!

**Technology**: web services, distributed systems, datamining, mobile communication, high performance computing

**Society**: democracy requirement, safety, security, environmental concerns

**Economics**: delocalization, outsourcing, complex supply chain, increasing weight of Finance

=>

New Problems? New way of setting old problems?
Taking into account interactions, safety requirements, economical stability.
Taking advantage from new technologies.
Old Problems/New Paradigms

=> Extensions toward New Contexts

- **Mixing Decision Levels:**
  Linking Routing, Pricing and Subsidizing.
  Linking Routing and Packing.

- **Non Standard Performance Criteria:**
  Robustness and Stochastic Complexity.
  Genericity.
  Reliability.

- **Non Standard Contexts:**
  Collaborative Planning.
  Dynamic Scheduling.
II LIMOS and Innovative Mobility

LIMOS, UMR CNRS/UBP 6158, CLERMONT-FERRAND

- MAAD: Decision Models and Algorithms
- SIC: Information and Communication Systems
- SP/ROGI: Production Systems, O.R for Industrial Engineering

+ Transversal Actions:
  - STIC-Mobility
  - STIC-Environment

- LABEX Participations:
  - Clervolc: Seismic/Volcanic Monitoring
  - IMOB3: Innovative and Intelligent Mobility
Partnerships: QUEBEC, HIT HARBIN, UT Compiègne, Centrale LILLE, SNCF…

Related Projects: Managing Decision inside New Generation Mobility Services

Context: need for more flexibility
- Growth of oil prices
- Environmental issues; City congestion?
- Increasingly old population.

Demand: mixing reactivity with mutualization, taking into account multimodality
New Generation Mobility Services

- CYCAB/VIPA Real Time Dial/Ride
- Ad Hoc Shuttle Fleet management
- AUTOLIB-Vehicle Sharing Intelligent Design and Monitoring
- Intelligent Co-Transportation Systems
- Internal Logistics Optimization
1. Cycab Dial/Ride/ Cycab Autolib

- Client Web
  - Formulaire de réservation des véhicules Cycab
- Serveur
  - La gestion et la sauvegarde des traces des réservations des véhicules Cycab
- LiveNode GPS/ZigBee
- État courant des CYCAB
  - Archivage
- Traitement des réservations
  - Acceptation/Refus
  - Instantané/Différé
- Liste des réservations courantes
- Archive
  - Réservation
  - État CYCAB
  - MAJ de l'état des CYCAB
- Pilotage des CYCAB
  - Envoi commande de réservation
  - Recevoir l'état du CYCAB
- Commande de réservation
- Requête de réservation acceptation/refus
- États CYCAB
  - Coordonnées GPS
  - En panne/en service
  - Fin de mission
- Exécution des commandes de réservation
- Signaler l'état de CYCAB
- Coordonnées GPS
LIMOS: Innovative Mobility

- Used on short distances
- Integration into the intermodal transport of the future

Target
- Large Parking Lots
- Large Factories
- Airports, Train Stations
- Hospitals, Campuses
- Business Centers
LIMOS: Innovative Mobility

- 2 Criterions
  - Economic (for the operator):
    - Number of used vehicles and total distance,
    - VIPA Load Rate,
    - Number of accident (reliability),
  - Service (for the user):
    - Connection speed,
    - Connection success rate (reliability).
2. AUTOLIB-Vehicle Sharing Intelligent Design and Monitoring

VIPA Fleet for AUTOLIB System ->
Relocation through wireless convoys

Input: Expected Demand Space-Time Distribution
Output: Relocation Strategy.
  - Relocation Signal: When? 
  - Relocation Process: how many convoy leaders?
    Process Duration? Convoy Routing?
    Convoy Making? Inter-Convoy exchanges?

Analogy with Ambulance Relocation
(Gendreau, Brotcorne, Laporte, Semet (2003, 2004))
3. Co-Transportation Systems

- **Usagers**: drivers and/or passengers
- **Vehicles** (driver owned):
  - Simple
  - Synchro (ad hoc communication
    Traceability devices)
- **Socio-economic players** (subsidizers)
- **Servicer**: intelligent web site +
  - Geolocalization/Mobile Com.
  - Infrastructure.
A Reference Problem: the Dial and Ride Problem (DARP).

**Input.**

- $V$: Vehicle set; $v \in V \rightarrow C(v) =$ Capacity characteristics, $S(v) =$ Speed characteristics, $\Delta(v) =$ Availability
- $K$: Object Set; $C(v)$ is a constraint on integer valued $K$-vectors
- $X$: Demand Set; $x \in X \rightarrow (o(x), d(x)) =$ origin/destination pair, $T(x) =$ Time Window, $D(x) =$ Load = Integer valued $K$-vector
- $G = (N, A) =$ Transit Network; $M =$ Related Shortest Path Distance Matrix

**Output.**

$v$ in $V \rightarrow$ a *timed* route $\Gamma(v)$: every node $s$ in $\Gamma(v)$ is provided with arrive-time, leave-time: time-space, load and unload: $X$
A Reference Problem: DARP

Output.
v in V -> a *timed* route \( \Gamma(v) \): every node \( s \) in \( \Gamma(v) \) is provided with arrive-time, leave-time: time-space, load and unload: \( X \)

Constraint.
- **Capacity constraint**: at any instant \( t \), current load \( L(v, t) \) of vehicle \( v \) compatible with capacity constraint \( C(v) \)
- **Time windows constraint**: any instant demand \( x \) is loaded and unloaded according to \( T(x) \)
- **Availability Constraint**: running time of vehicle \( v \) is included into \( \Delta(v) \)
- **Speed constraints**: for any vehicle \( v \), any consecutive nodes \( s_1, s_2 \) in \( \Gamma(v) \), arrive-time(\( v, s_2 \)) – leave-time(\( v, s_1 \)) is compatible with \( M \) and \( S(v) \)
- **Load/unload time constraint**: for any vehicle \( v \), any node \( s \) in \( \Gamma(v) \), leave-Time(\( v, s \)) – arrive-Time(\( v, s \)) compatible with \( T(x) \) and \( D(x) \), \( x \) loaded and unloaded in \( s \).
Performance.
Mix (Multicriterion) = \text{Card}(\text{Active-Vehicle}), \sum_v \text{Length}(\Gamma(v)), \sum_x \text{Duration}(\Gamma, x))

Extensions
- **Vehicle Preemption**: a demand \( x \) may be routed from \( o(x) \) to \( d(x) \) through several vehicles.
- **Load Preemption**: the load \( D(x) \) may be split into several sub-loads, which are routed independently.

Static Versus Dynamic.
- **Static**: all data are known in advance;
- **Dynamic**: data come as a dataflow; current roadmap of every vehicle is taken into account.

Remark: most often, time windows flexibility is maintained
IV. Standard Methods and Benchmarking.

**Simplified Framework**: Nodes are splitted according to the demands: any $o(x), d(x)$ is identified with a specific node.

**A Simple MIP model (Not Practical...!)**.

**Variables**
- $t = (t_n, n \in N)$, rational, $z = (z_{nm}^v, n, m \in N, v \in V)$ with $\{0, 1\}$ values:
  - $z_{nm}^v = 1$ means arc $(m,n)$ is part of route $\Gamma(v)$
  - $t_n = \text{time at node } n$, identified with some demand load/unload, is “serviced”
  - $p_n = \text{load at node } n$, for the vehicle $v$ which services $n$

**Constraints**:
- $z$ represents a partition of $N$ into circuits (Tour Constraints)
- $\sum_v z_{nm}^v = 1 \Rightarrow t_m - t_n \leq M(n, m)$ (Logical Time Constraints)
- $\sum_v z_{nm}^v = 1 \Rightarrow p_m - p_n = \text{Load}(n)$ (Logical Load Constraints)
- $t_{o(x)}, t_{d(x)}, t_{d(x)} - t_{o(x)}$ inside related time windows;
- $p_{o(x)}$ compatible with capacity constraints.

**Goal**: Minimize Cost.$z + \sum_x t_{d(x)} - t_{o(x)}$.
**Greedy Insertion Scheme:**

JAW (86), Xiang, Xu, Chen (2006), Toussaint/Quilliot (2010)
Demands are randomly ordered, and inserted according to this order into current partial routes \( \Gamma(v), v \in V \)
(filtering process through constraint propagation)

**Local Search and Metaheuristics scheme (Tabu, Simulated Annealing...)**

- Calvo, Colorni (2006): Heuristics Insertion/Assignment
- Psafaratis, Sexton, Bodin (80, 79, 85, 95)

**Local operators:**

- **Exchange:** 2 demands are exchanged between 2 tours
- **Shift:** 1 demand is shifted from one tour to another one;
- **Internal-Shift:** 1 demand is relocated inside a given tour
A Reference Problem: DARP

Branch/Bound, Branch and Cut:

Ropke, Cordeau, Laporte (2001, 2003);
Exact results up to 25 demands
Branching Process: on the variables $z^{v}_{nm}$ of the PLNE representation
Bounding process: using the PLNE representation + ad hoc cuts

Toussaint/A.Q (2010)
Greedy Insertion + Branch/Bound
Branching Process + Constraint Propagation:
Demand $x$ in tour $v$?
Efficient if sharp time window constraints.
Dynamic Flow (Flow over Time):

**Recall: Flows/Multicommodity Flows**

Network $G = (Z, E)$

**Flow** $z = E (arc)$ indexed vector such that for every node $x$,

$$
\sum_{e \text{ enter } x} z_e = \sum_{e \text{ out } x} z_e,
$$

(*Kirshoff Law*)

Kirshoff Law may be adapted in order to make $z$ express

the routing of a given quantity from one node to another

**Multicommodity Flow**: collection of flow vectors, whose values identify distinct class of objects

**Dynamic Flow Framework**: nodes are (pair $(n, t)$, n in N, t in the time space) -> Explicit or implicit representations

**DARP**: Vehicle Flow + Multicommodity Demand Flow, tied with coupling capacity constraints (Master/Slave scheme) -> Local operators related to the flow/multicommodity-flow machinery -> Cancelling circuits/cycles


Column Generation: main vector indexed on the set of all the possible tours -> induced subproblem: Generating efficient tours.

Clustering: master vector index on the set of X subsets, i.e: which demands are handled by the same vehicles; slave object: the tour related to some subset A of X, which is part of the cluster. Column generation -> generating the ad hoc subsets A.
**A Reference Problem: DARP**

**Dynamic Context:** (few studies)
- BERLIN-TELEBUS, Bjorndorfer 97: Extraction of Seed Trajectories
- Madsen, Rygaerd, Ravn (Copenhagen TAD System, 1995): Adaptation of Jaw Insertion Techniques
- Todorovic, Radjonoivic (2000): Application of Fuzzy Logic Rules
- Colorni, Righini (2001): Real Time Clustering through Local Transformation
Remarks about usual dynamic models.

Models: dataflows, algorithms take into account current roadmaps of the vehicles;

Demands: n = 25 to 900, no focus on the real time constraints induced by communication and supervision;

What about taking into account stochastic demand distribution?

Soft management of real constraints: time windows remain open all throughout the process, until the user is serviced.

What about system/user communication and « rendez-vous » mechanism?

Dynamic most often means « perturbation handling »: what about failure (vehicle delay, user give up…)?

A Reference Problem: DARP
A Reference Problem: DARP

Static/Dynamic DARP: Benchmarking.

TSP LIB, Laporte Cordeau Instances, ... -> Toy Problem -> many academic test beds

A few word about instances generation:

**Fagin Theorem:** The theoretical values of randomly generated instance with non null density \{0, 1\} of a problem expressed according to the 2 order monadic logic formalism converge almost surely (either to 1 or to 0).

**Courcelle Theorem:** 2 order monadic logic problems with bounded clique width are time-polynomial
A Reference Problem: DARP

In most cases, testbed instances => generated according to the 2 order monadic logic formalism.

=> They are strongly biased.

Experiments -> Testing ad hoc Indicators on common testbeds:

⇒ parallelism rate: number of demands which may be simultaneously handled;
⇒ Dispersion rate: variance of o(x), d(x) distribution....

We remark: very concentrated distribution.
⇒ Generating meaningful testbeds is a difficult game.

Example: Cordeau/Laporte instances -> very strong temporal constraints -> getting initial solution is difficult -> advantage to constraint propagation + “repairment” heuristics.

Dynamic Case: what about dataflow generation, and launching of the recomputation process? What has to be measured?
V. DARP: New Contexts mean New Paradigms.

- **Mixing Decision Level** (routing/packing, economical management)
- **Non Standard Criterion** (reliability, robustness…)
- **Non Standard Contexts** (collaborative, reactive…)

DARP: New Paradigms
**V.1. Mixing Decision Levels.**

**Linking Routing and Packing:** loads are 2D or 3D-objects, with geometrical characteristics -> Non trivial testing of capacity constraints, time consuming loading and unloading operations -> 3L-CVRPV (Duhamel, Lacomme, Quilliot, Toussaint)

**An approach:** introducing learning devices (SVM, Neural Network..) in order to deal with the *weak* and *strong* feasibility of 2D and 3D-packing

**At stake:**
- Simultaneously dealing with distinct granularity levels;
- -> Getting fast approximation results for complex problems
- Ensuring consistency of linked models -> Getting fast approximation results for complex problems
DARP: New Paradigms

Computation of tours and packing

Depot

Vehicle 1

Vehicle 2

Vehicle 3

1

2

3

4

5

6

7

Tour 1

Tour 2

Tour 3

x3
DARP: New Paradigms

Linking Routing/Pricing/Subsidizing.

Routing policy + **Expected Demand** -> Expected Costs
Prices + Routing Policy -> QoS -> **Expected Demand**
**Question:** which prices, which subsides?

**Approach:**


(2). **Master Slave (bilevel) Decomposition Scheme:**
- **Main Problem** -> Prices
- **Slave Problem** -> Designing a routing policy for a user subgroup
- **Technological Gap:** *Evaluating Price/QoS Elasticity of Demand*
V.2. Non Standard Criteria.

**Robustness/Stochastic Complexity.**

*At stake:* Adaptability of the solution when it comes to implementation.

*Example:* DARP (dynamic/Static) $\Rightarrow$ Current Schedule $\sigma$: which ability to take into account future demands, unexpected delays and “rendez-vous” failure?

*Difficult problem:* Currently suffering from a deficit of formal approach.
**The basic point:** the problem cannot be handled according to its current representation

- **Input data:** must involve a formal and quantified representation of the events: ad hoc language;
- **Output Object:** must take the form of:
  - A set of constraints and decision rules;
  - A strategy (decision tree) on those constraints

**Example:** Simple DARP

**Schedule:** a set $\Lambda$ of additional constraints: (Un)Load(x) precede (Un)Load(y) on vehicle $v$ + implicit priority rule.

**Schedule Strategy** (mixed schedule): set of decision rules.

- **Rule:** Instant $t$, State $S$ contain pattern $E$
  - Finished tasks $A$, Currently running task $B$ $\models$ Modify $\Lambda$

**A problem:** part the schedule language semantics must be shared by the users.
DARP: New Paradigms

**Reliability.**

**VIPA DARP:** Avoiding “hazardous” manoeuvring: overtaking..., avoiding schedule modifications

=> Making passenger of a given vehicle share same loading and unloading nodes.

**At stake:** conveniently modelling reliability in a given monitoring context, and casting it into the decisional framework.

DARP: New Paradigms

- Stops for users and maintenance
- A one-way loop with outputs for stations
- Homogeneous fleet of autonomous vehicles (VIPA)
- Users ask for a vehicle via mobile phone or a terminal
DARP: New Paradigms

Diagram:
- **DBMS**: Update VIPA’s state
- **Listener**: Save and update queries
- **Terminal**: Forward Queries
- **Users**: Send Location & Batterie life
- **Supervisor**: Share data, Save roadmap, Broadcast roadmaps
- **VIPA**: Go In, Go Out

LIMOS
DARP: New Paradigms

Genericity.

DARP Contexts: Highly Evolutive, Continuum Dynamic/Static
At stake: Development cost, adaptability to model evolution.

Generic Framework?
- Dynamic Flow/Time Over flow
- Ruled Based Systems
- Insertion Algorithms...
DARP: New Paradigms


Taking into account technological, organizational, societal context!

**Collaborative Planning**

The principle: even if you are the “boss”, negotiation is at the heart of any decision process.

**DARP**? The ruler of a DARP service may not be in direct control of all the vehicles involved in the system: mix of AUTOLIB shared vehicle fleet, ad hoc shuttle fleet, “co-transportation” devoted individual cars -> Dependence on the will of other players (subcontractors), which have their own agenda and criteria.
DARP: New Paradigms

An illustration of Collaborative Planning: The Doodle.

A “master”, and its partners => May be viewed as a collaborative RCPSP.
Main task: the meeting; Auxiliary tasks: the moves of the partners
Partners are at the same time resources and tasks.

Decision oriented computing devices (and related models):
- A master device $\mathcal{M}$: consider the constraints provided by the partners and schedule the meeting;
- Partners devices $\mathcal{P}_i$, $i = 1..N$: schedule partner $i$, compute constraints and transmits them to $\mathcal{M}$;

Process Main Loop:
$\mathcal{P}_i$, $i = 1..N$ $\leftrightarrow$ $\mathcal{M}$: $\rightarrow$ succeed or fail in computing
accept or reject the proposal

The master $\mathcal{M}$: if failure, ask some of the $\mathcal{P}_i$, to relax their constraints Else send the proposal to the partners
The partners: \( \mathcal{P}_i, i = 1..N \):
- If they reject the proposal, send new constraints to \( \mathcal{M} \) Else: OK.

Requirements: Design a common constraint language: syntax/semantics
- Design \( \mathcal{P}_i \), in such a way they compute constraints (cf Robustness)
- Handling hidden part…! It is like playing a game. Not everybody want the same thing. Ex: partner j may want the meeting without partner k.

A theoretical framework: Pricing:
- Master M, schedule \( \sigma \) -> Value \( V(\sigma) \) resulting from model \( \mathcal{M} \);
- Partner i, schedule \( \sigma \), Value \( V_i(\sigma) \), resulting from model \( \mathcal{P}_i \);

Questions:
- Which payments between M and its partners in order to make possible reaching a convenient schedule?
  - Cooperative Game Framework (Shapley, Core notion…)
  - Concurrential (non cooperative) framework? (Nash…)
DARP: New Paradigms

Dynamic Scheduling.
- Real Time DARP

Customer x (tasks) asks for service from origin node o(x) to destination node d(x), while imposing temporal constraints.
Innovative Mobility

Decision-making

[A0 Loop]

[Fleet to Optimize]
Run A1

A0 Loop

Unstack A0 Loop

A1
1 - Static
2 - Dynamic
3 - Simulation

Share Data

DBMS

VIPA

Broadcast roadmaps

Share Data
**DARP: New Paradigms**

**Process:**

Between instant $t_{n-1}$ and instant $t = t_n$, customers ask

(the supervisor or some of the vehicles: centralized/decentralized) for service;

Instant $[t, t+\alpha]$: some activation process $A$ decide to launch the replanification process $P$,

Instant $[t+\alpha, t + \alpha + \beta]$: $P$ compute a new planning for the vehicles, send answers to the customers: meeting proposal or rejection of the demand, and send orders to the vehicles;

Instant $[t+\alpha+\beta, t_{n+1}]$: vehicles and customers run their way, new demands are registers, as well as failed meetings or rejected proposals.

**Requirements:**

- Design algorithmic processes A and B; Models: which meaning to “replanification”, acceptable for users and communication system
- Adquire and conveniently model input data;
- Evaluate.
DARP: New Paradigms

The basic points:
- The stochastic dimension of the problem cannot be ignored;
- A priori evaluation must be performed while considering that the input is a stochastic process;
- A posteriori evaluation (test) must be performed through **simulation**;
- The underlying decisional model (module $\mathcal{D}$) must take into account:
  - *QoS criterion related to the meetings* (waiting times…)
  - *Safety concerns related to communication process between the systems, the vehicles and the customers* (ensuring the reliability of the meetings).

Consequence: the decisional model becomes very different from the static one, and not only a “on line” adaptation of this static model
VI. Conclusion.

O.R: a risk of getting old…

New Trends: arise from societal and technological change

But: Tackling new issues requires more than inserting additional constraints and applying old processes.

Smart cities: a very rich play-ground