

Optimization of strategy planning for building deconstruction

Eva Queheille, Franck Taillandier, Nadia Saiyouri

▶ **To cite this version:**

Eva Queheille, Franck Taillandier, Nadia Saiyouri. Optimization of strategy planning for building deconstruction. Automation in Construction, 2019, 98, pp.236-247. 10.1016/j.autcon.2018.11.007 . hal-02936143

HAL Id: hal-02936143
https://hal.inrae.fr/hal-02936143

Submitted on 18 Feb 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Optimization of strategy planning for building deconstruction

Eva Queheille*, Franck Taillandier, Nadia Saiyouri

University of Bordeaux, UMR CNRS 5295 I2M, 33405 Talence, France

Waste management is a growing concern in the construction sector and particularly within demolition works. In order to encourage better management, the European Union has set up a minimal 70% recovering rate - in mass -for building and public works by 2020. And to help deconstruction companies to reach this aim without extra cost or delays, a Multi-Objective Optimization algorithm is suggested. The algorithm will work out a decon-struction strategy - i.e. type of waste treatment, number of required workers… - and will take into account three objectives - cost, delay and recovering rate - with 17 decision variables. Use on a real deconstruction situation highlights the interest of the algorithm. Eight optimization algorithms were tested and the most efficient al-gorithm is Decomposition-Based Evolutionary Algorithm. It came up with better strategies than the company one. This work proves that environmental performance can be improved without increasing cost or delay.

Table 1
Abbreviations table.

Abbreviation	Definition	Value range	Variable type
C	Deconstruction work cost	Calculated during optimization	Not a variable
C[…]	Work phase cost […], e.g. dismantling, loading of dismantling waste	Calculated during the optimization	Not a variable
D	Deconstruction work duration	Calculated during the optimization	Not a variable
D[…]	Work phase duration […], e.g. dismantling, loading of dismantling waste	Calculated during the optimization	Not a variable
DC[…]	Daily cost of the resource […], e.g. worker, mini-excavator	Propriety of the industrial partner	Not a variable
Dem	Building demolition phase	None	Not a variable
Demol	Demolition process choice	0: Demolition and loading are carried out successively 1: Demolition and loading are executed in the same time	Decision variable
Dis	Building dismantling phase	None	Not a variable
Dism	Dismantling process choice	0: Dismantling done by workers and/or mini-excavators 1: Dismantling undone 2: Partial dismantling proceeded by excavators, at the same time as demolition phase	Decision variable
E_{Dem}	Excavators, with drivers, for demolition phase	1	Not a variable
E_{LDem}	Excavators, with drivers, for loading of demolition waste	1	Not a variable
ER[…]	Expansion rate of waste […], e.g. inert waste, wood waste	Propriety of industrial partner	Knowledge variable
FL_{LDIS}	Front-end loaders for loading of dismantling waste phase	1	Not a variable
I	Work plant installation phase	None	Not a variable
L[…]	Loading of waste from deconstruction phase […], dismantling or demolition	None	Not a variable
M[…]	Mass of waste […], e.g. concrete, plaster	Estimated before optimization	Case study variable
ME_{Dis}	Mini-excavators, with drivers, for dismantling phase	0 to 3	Decision variable
R[…]	Waste recovering rate depending on type of dismantling and on chosen treatment […], e.g. recovering of concrete with dismantling done	Estimated before optimization	Knowledge variable
RD[…]	Daily truck turns carried out for type of waste […], e.g. inert waste, wood waste	Propriety of industrial partner	Knowledge variable

* Corresponding author.
E-mail address: eva.queheille@u-bordeaux.fr (E. Queheille).

Abbreviation	Definition	Value range	Variable type
SF_{Dem}	Site foremen for demolition phase	1	Not a variable
SF_{Dis}	Site foremen for dismantling phase	0 to 2	Not a variable
SF_{I}	Site foremen for installation phase	1	Not a variable
Ship[…]	Shipping cost of the machine resource […], e.g. excavator, front-end loader	Propriety of the industrial partner	Knowledge variable
TreatC[…]	Treatment plant cost per ton for waste […], e.g. recovering plant for concrete	Propriety of the industrial partner	Knowledge variable
Treat[…]	Chosen treatment plant for waste […], e.g. recovering plant for concrete	0: Landfill 1: Sorting plant 2: Recovering plant	Decision variable
UM_{Inert}	Container volume, then transformed in mass for calculation, for evacuation of inert waste	$12\,m^3$, $15\,m^3$ with a simple container or $15\,m^3$ with a dumper truck	Knowledge variable
$UV_{Non\text{-}hazardous}$	Container volume for evacuation of non-hazardous waste	$12\,m^3$ or $30\,m^3$	Knowledge variable
UV_{Wood}	Container volume for evacuation of wood waste	$12\,m^3$, $30\,m^3$ or $60\,m^3$	Knowledge variable
V[…]	Volume of waste […], e.g. concrete, wood	Estimated before the optimization	Case study variable
W_{Dem}	Workers for demolition phase	1	Not a variable
W_{Dis}	Workers for dismantling phase	0 to 10	Decision variable
WE[…]	Work efficiency of a resource for work phase […], e.g. dismantling by workers	Propriety of the industrial partner	Knowledge variable
W_{I}	Workers for installation phase	2	Not a variable
W_{LDem}	Workers for loading of the demolition waste	1	Not a variable
W_{LDis}	Workers for loading the front-end loader to evacuate the dismantling waste	0 to 3	Decision variable

1. Introduction

Waste management is a major environmental issue, bringing on energy consumption, pollution and loss of agricultural lands. These problems do not refer only to household waste, but also building waste. Demolition and rehabilitation waste represent about 35 millions of tons per year in France [1] and will continue to increase in the following years. In order to encourage a better building waste management, the European Union has set up the objective of a minimal 70% recovering rate - in mass - for waste from building and public works by 2020 [2]. France, as a member of the European Union, has to fulfill this objective within its territory. Main buildings in France are made of concrete and the legislative recovering rate can be assumed easy to obtain if concrete is reused or recycled. Nevertheless, recyclers' requirements, environmental objectives of buildings owners and laws increase while cost and delays granted for demolition works decrease. It has a significant impact on demolition companies and their studies. For instance, waste sorting on site often appears as too expensive to demolition companies whereas it is a real add-value [3].

In order to reach the European Union recovering objective, software has been created to help decision with building waste treatment. Most programs assess waste amount and help decision for waste transport and treatment [4–6], with sometimes a regional map of treatment plants [7,8]. Other programs identify attractive treatment plants according to the site location and waste that will be disposed of [9]. These programs have a common point: they all begin after deconstruction, when waste is already produced. However, waste management depends on the whole process of deconstruction. Deconstruction resources and the level of sorting on site affect quality of salvaged materials and available treatments. For instance, if materials of a wall - structure, insulation, coating - are not separated and sorted, it is not possible to recycle them. Deconstruction strategies come under issues like cost, time and building type. Waste type and amount are linked to the building type. In the building sector, numerous types of waste exist - concrete, bricks, stones, tiles, glass, wood, plaster, metals, isolation wool…-, which brings about stronger difficulty for deconstruction projects preparation.

Few programs help demolition companies for deconstruction planning. Buhler et al. [8] work on waste treatment with different sorting levels, but only three are predefined in the software. Schultmann et al. [10] developed a software of a wider scale. Unlike other programs, it calculates waste amount from the building elements, that the user input in architectural plans. It is a forerunner way to Building Information Modelling (BIM). With these data, the software gathers waste by type and the user selects the treatment for every waste family to obtain the treatment plan cost. The software estimates cost of deconstruction works by helping the user plan deconstruction equipment, that it is especially innovative. Despite these features, programs need a massive user implication and, for most of them, cannot help him to explore solutions beyond his own experience. The user has to enter volume, mass, waste density, treatment plants prices, then choose treatment for each waste, put in order every deconstruction task… Few programs guide the user to (i) an unheard-of solution, e.g. a new treatment, (ii) another better method than the usual one, e.g. carry out a bigger sorting on plant or (iii) integrate environmental purpose. We assume that a software needs to help the demolition company to build, assess and compare strategies with more classic demolition strategies. Thus it should favor environmentally friendly solutions without degrading deconstruction cost or delay.

In this paper, deconstruction planning is modelled as a multi-objective problem - part 2 - in order to build strategies more environmentally responsible while allowing for economic and duration feasibility. The algorithm received the expertise of Bordeaux Démolition Services (BDS), a French company specializing in building deconstruction. BDS provided the case study on which the software was tested - part 4. The computation problem was modelled in a demolition company point of view. In order to solve the multi-objective problem, several algorithms are compared in part 3 - NSGA-II [11], OMOPSO [12], CMA-ES [13], PAES [14], SPEA2 [15], DBEA [16], MOEA/D [17] and VEGA [18]. We assessed their relevance to bring out the most appropriate algorithm for the deconstruction optimization problem.

2. Modelling deconstruction as a multi-objective problem

2.1. Demolition process

A typical deconstruction work can be summed up as the reverse of construction. The last element set up during the building construction is the first element removed during the deconstruction. The work is carried out in successive phases. During first one, “Installation”, the site is prepared before deconstruction works. It includes installation of fence, barriers, electricity and water supply… Then dismantling phase begins. Workers and machines remove elements inside the building, such as furniture, drywalls, windows, doors, electrical equipment… Elements are loaded into containers by workers or loaders. Container volume depends on the estimated amount of waste. For some deconstruction projects, dismantling phase is skipped: the building is directly demolished without dismantling, which is faster but limits recovering. When the dismantling phase is finished or skipped, an excavator demolishes the structural part of the building, from top - the roof - to the bottom – the foundations. Then the excavator loads waste in containers or directly in trucks. Deconstruction work is closed when the site is cleaned and waste is evacuated to treatment plants.

2.2. Objectives and variables

2.2.1. Objectives

The multi-objective problem objectives were chosen from the point of view of a demolition company stakeholders. Three objectives were set up:

- Work duration, in number of days
- Cost, in euros, for the deconstruction company to do the job, including waste management
- Recovering rate, in mass percentage.

Cost and duration objectives are classic in any project. The recovering rate represents an environmental criterion in deconstruction works. It helps to compare strategies with the European Union objective, which is to recycle at least 70% of waste mass by 2020 [2]. The recovering rate will increase in the following years and is still a commercial point for demolition companies.

The objectives calculations are described in the Sections 2.2 to 2.4. The three objectives are computed from a set of variables. Three kinds of variable are identified: decision variables, knowledge variables and case study variables.

2.2.2. Decision variables

In our algorithm, the objective calculation contains 17 decision variables. The set of decision variables values makes a deconstruction strategy. The goal is to adjust the values in order to optimize the deconstruction strategy. Decision variables refer to five domains: dismantling process, demolition phases, resources - human and mechanical -, waste containers and waste treatment.

The first decision variable *Dism* gives a choice for the dismantling phase process. Dismantling is about removing the inside of the building before demolishing the building with demolition phase. Three choices have been studied:

- Dismantling done by workers and/or mini-excavators (*Dism* = 0)
- Dismantling undone, which means that works begin directly with demolition by excavators (*Dism* = 1), without previous dismantling
- Partial dismantling done by excavators, at the same time than demolition phase (*Dism* = 2).

In deconstruction works, dismantling can be seen as a long and expensive process. With this decision variable, we would be able to confirm or not if a dismantling process is relevant for deconstruction works.

The second decision variable *Demol* refers to the demolition phase. Two different processes are possible. In the first process, one excavator demolishes the structural part of the building, then loads waste for evacuation. In the second process, two excavators work at the same time: one excavator for demolition and another one for loading. Then the variable *Demol* will have a significant impact on work duration:

- Demolition and loading are realized successively (*Demol = 0*). One excavator achieves the two sub-phases.
- Demolition and loading are executed in the same time (*Demol = 1*). Two excavators work in the same time on the site.

The third set of decision variables refer to human and mechanical resources. In this paper, only the most used resources for dismantling and demolition phases are used. Each phase is separated in two sub-phases: the deconstruction work and the loading of waste. For the demolition phase (*Dem*), resources do not represent decision variables because their value is set up by the decision variable *Demol*. For the dismantling phase (*Dis*), several decision variables represent resources:

- Workers for dismantling (W_{Dis})
- Mini-excavators for dismantling (ME_{Dis})
- Workers for loading the front-end loader in order to evacuate dismantling waste (W_{LDis}). The front-end loader (FL_{Dis}) is by default

Table 2
Set of decision variables for a deconstruction strategy.

Set of decision variables	Decision variable	Possible values
Dismantling process	Dismantling (*Dism*)	{Yes, no, during demolition} with values [0–2]
Demolition process	Demolition (*Demol*)	{Successively, At the same time} with values [0–1]
Human and mechanical resources for dismantling	Workers (W_{Dis})	[0 − 10]
	Mini-excavators (ME_{Dis}) and its drivers (MED_{Dis})	[0–3]
	Workers for loading the front-end loader (W_{LDis})	[0–3]
Volume of the waste containers	Inert waste container (UM_{Inert})	{12 m^3, 15 m^3 with a simple container, 15 m^3 with a dumper truck} with values [0–2]
	Wood waste container (UV_{Wood})	{12 m^3, 30 m^3 or 60 m^3} with values [0–2]
	Non-hazardous waste, except wood, container ($UV_{Non-hazardous}$)	{12 m^3 or 30 m^3} with values [0–1]
Waste treatment plant	Treatment plant for concrete ($Treat_C$)	{Landfill, sorting plant, recovering plant} with values [0–2]
	Treatment plant for glass ($Treat_G$)	{Landfill, sorting plant, recovering plant} with values [0–2]
	Treatment plant for mixed inert ($Treat_{MI}$)	{Landfill, sorting plant} with values [0–2]
	Treatment plant for wood ($Treat_W$)	{Landfill, sorting plant, recovering plant} with values [0–2]
	Treatment plant for metal ($Treat_M$)	{Landfill, sorting plant, recovering plant} with values [0–2]
	Treatment plant for plaster ($Treat_P$)	{Landfill, sorting plant, recovering plant} with values [0–2]
	Treatment plant for plaster bricks ($Treat_{PB}$)	{Landfill, sorting plant, recovering plant} with values [0–2]
	Treatment plant for furniture ($Treat_F$)	{Landfill, sorting plant, recovering plant} with values [0–2]
	Treatment plant for mixed non-hazardous ($Treat_{MNH}$)	{Landfill, sorting plant} with values [0–2]

chosen as one.

In our algorithm, human resources are reflected by an exact number of individuals, and not by team. Team could have been chosen to ease the algorithm. However, work efficiency is known per individual or machine. The partner company prefers to set up the resource planning with individuals rather than with crews. We chose to consider workers as individuals and not teams in order to be closer to the partner practices. This choice has a drawback: it requires to make a strong assumption on the work rate - which is broken down to the unit by worker - and to assume a linear work rate. But an advantage can be highlighted: it allows the study engineer to define the size of the team - workers, machines… - in accordance with the size of the building to demolish.

With the fourth set of decision variables, volume of waste containers is chosen to calculate the number of necessary truck turns to evacuate the containers when they are filled. Then cost for waste transport will be calculated. Further explanations are given in Section 2.4.

The last set of decision variables outlines waste treatment choices. The most classic waste types have been studied. They have been separated with the two major deconstruction phases:

- Dismantling waste: glass, wood except timber frame and wooden floor structure, metals, plaster, bricks with plaster, furniture, other non-hazardous waste…
- Demolition waste: building structure like concrete or other inert waste, timber frame and wooden floor structure…

The recovering rate depends on the chosen treatment. The three treatments in this study are: Landfill ($Treat = 0$), Sorting plant ($Treat = 1$) and Recovering plant ($Treat = 2$). Recovering includes reuse and recycling treatments. In France, reuse is mainly for concrete or furniture. However, reuse of concrete directly on the site is really uncommon; it is at best recycled for road construction. This recycling is assumed as recovering.

One treatment plant is chosen for each waste. In the algorithm, great attention has been paid to treatment feasibility. In fact, for some waste, e.g. mixed non-hazardous waste, direct recovering - direct evacuation to a recovering plant - is not possible. Furthermore, treatment feasibility also depends on the choice of dismantling process. If deconstruction works are resumed in simple demolition ($Dism = 1$ or 2), only excavators would be present in the plant. With this machine, waste sorting is a difficult task and then less waste can be evacuated directly in a recovering plant. This point is modelled as a constraint – see Section 2.6.

Table 2 sums up the 17 decision variables and their possible values. The variables boundaries prevent creation of impossible strategies, such as 50 dismantling workers on site. While changing boundaries, we adapt the algorithm with the building and the site. For example, if the site is not wide enough for two excavators, the decision variable *Demol* is restrained to the value $Demol = 0$.

2.2.3. Knowledge variables

Knowledge variables gather all the required data to run the objectives calculation. They are constant in a given context. Knowledge variables refer to:

- Resource efficiency to perform a task, e.g. dismantling manually, loading waste with an excavator, loading waste with a front-end loader… These values are necessary to estimate work duration - first objective -, but also global cost - second objective.
- Resource cost per day, that contribute to the global cost of the deconstruction work - second objective.
- Treatment price per ton of waste, depending on the waste and the contract that links the treatment plant and the deconstruction company. These prices are also involved in the global cost of the deconstruction work - second objective.

Data can be adjusted with the deconstruction company or even local prices for waste treatments. Data used for this study belong to BDS company and their contracts with treatment plants. They cannot be published in this paper.

2.2.4. Case-study variables

Finally, case-study variables depend on the building to demolish. It integrates:

- Ground surface of the building, in square meters
- Developed surface of the building - sum of each inhabited level surface - in square meters
- Length of necessary fence to close the site, in meters
- Mass - in tons - and volume - in cubic meters - of each waste. In order to ease the study, hazardous waste, e.g. asbestos, have not been taken into account.

2.3. Work duration objective

Work duration (D in equations) is calculated, in number of days, for all the phases set in the Table 3. Duration depends on chosen human and mechanical resources; each resource has an appropriate work efficiency.

Installation phase includes works such as installation of fence to close the plant, road protections… Thus, installation phase depends on the site size and the team formed for these tasks. The duration (D_I) is calculated as follows (Eq. (1)). In the company partner, one day of installation is always attributed for common deconstruction works. For more important works where safer disposition are necessary, e.g. barriers on windows or on the roof, the work efficiency would need to be changed.

$$D_I = working\ site / WE_H \quad (1)$$

Dismantling duration depends on the use of machines - mini-excavators - and number of workers. In our algorithm, mini-excavators can only be attributed to dismantling of the buildings ground levels. In fact, moving machines in the superior levels depends on buildings access and floor resistance. For calculation, dismantling duration (Eq. (4)) is separated in two parts: ground surface (Eq. (2)) in which machines can be used, and upper levels (Eq. (3)) demolished manually.

Table 3
Work efficiency depending on chosen human and mechanical resources.

Deconstruction phase	Work efficiency of the associated resources
Installation (*I*)	Human (WE_H), per worker
Dismantling (*Dis*)	Manual (WE_W), per worker
	Mechanical (WE_{ME}), per mini-excavator
Loading of dismantling waste (*LDis*)	Manual with manual loading (WE_{LW}), per worker
Demolition (superstructure and infrastructure) (*Dem*)	Mechanical for superstructure if $Dism = 0$ or 1 ($WE_{SE0,1}$), per excavator
	Mechanical for superstructure if $Dism = 2$ (WE_{SE2}), per excavator
	Mechanical for infrastructure (WE_{IE}), per excavator
Loading of demolition waste (*LDem*)	Mechanical (WE_{LE}), per excavator

$$\begin{cases} if\ Dism = 0 \Rightarrow D_{Dis1} = Ground\ surface/(WE_W \times W_{Dis} + WE_{ME} \times ME_{Dis}) \\ if\ Dism = 1\ or\ 2 \Rightarrow D_{Dis1} = 0 \end{cases} \quad (2)$$

$$\begin{cases} if\ Dism = 0 \Rightarrow D_{Dis2} = (Developed\ surface - ground\ surface) \\ \qquad\qquad /(WE_W \times W_{Dis}) \\ if\ Dism = 1\ or\ 2 \Rightarrow D_{Dis2} = 0 \end{cases} \quad (3)$$

$$D_{Dis} = Rounded\ up\ (D_{Dis1} + D_{Dis2}) \quad (4)$$

Produced waste with dismantling process is loaded in containers with a front-end loader. The loader is loaded itself by workers, and the number of workers influence the loading speed (Eq. (5)).

$$\begin{cases} if\ Dism = 0 \Rightarrow D_{LDis} = Rounded\ up\ (Dismantling\ waste\ volume \\ \qquad\qquad /(WE_{LW} \times W_{LDis})) \\ if\ \ Dis = 1\ or\ 2 \Rightarrow D_{LDis} = 0 \end{cases} \quad (5)$$

After dismantling, the building structure is demolished in the demolition process. Works are separated following the two parts of a building (Eq. (6)): (1) Superstructure - exterior walls, floor structure, structural frame and roof - and (2) Infrastructure - elements below the building, e.g. foundations. Demolition duration depends on the choice of the dismantling process. Work efficiency of excavators for superstructure demolition is the same for $Dism = 0$ or 1. However, if $Dism = 2$, it means that the excavator will demolish slower in order to do a better waste sorting than with a simple demolition.

$$\begin{cases} if\ Dism = 0\ or\ 1 \Rightarrow D_{Dem} = Rounded\ up\ [Developed\ surface \\ \qquad\qquad /(WE_{SE0,1} \times \ E_{Dem}) + Ground\ surface/(WE_{IE} \times E_{Dem})] \\ if\ Dism = 2 \Rightarrow D_{Dem} = Rounded\ up\ [Developed\ surface/(WE_{SE2} \times \ E_{Dem}) \\ \qquad\qquad + Ground\ surface/(WE_{IE} \times E_{Dem})] \end{cases} \quad (6)$$

As for dismantling process, demolition waste is evacuated (Eq. (7)). In this phase, loading is done by an excavator, because of the large sections of waste like foundation concrete. If the dismantling process has been carried out ($Dism = 0$), then the excavator just needs to evacuate waste from demolition process. Otherwise, if the dismantling process has not been realized ($Dism = 1$ or 2), totality of the waste has to be loaded in this phase.

$$\begin{cases} if\ Dism = 0 \Rightarrow D_{LDem} = Rounded\ up\ [Demolition\ waste\ volume \\ \qquad\qquad /(WE_{LE} \times E_{LDem})] \\ if\ Dism = 1\ or\ 2 \Rightarrow D_{LDem} \\ \qquad\qquad = Rounded\ up\ [(Dismantling\ waste\ volume \\ \qquad\qquad + Demolition\ waste\ volume)/(WE_{LE} \times E_{LDem})] \end{cases} \quad (7)$$

The duration objective needs to minimize the work duration (Eq. (8)). To determine duration of the whole phase of dismantling, we chose to take the longest duration between dismantling and loading of dismantling waste, assuming that these two works can be done in the same time without difficulties. It is even a necessity to get back space for the next dismantling task. If demolition and loading phases are executed in the same time, when $Demol = 1$, the longest duration between the two works is taken. Then, even if calculation gives a loading duration shorter than demolition duration, the excavator for loading will be present for the whole process of demolition, in order to evacuate with the same speed that the other excavator demolishes.

$$Minimize\ D\ with \begin{cases} if\ Demol = 0 \Rightarrow D = D_I + MAX\,(D_{Dis}; D_{LDis}) + D_{Dem} \\ \qquad\qquad + D_{LDem} \\ if\ Demol = 1 \Rightarrow D = D_I + MAX\,(D_{Dis}; D_{LDis}) \\ \qquad\qquad + MAX\,(D_{Dem}; D_{LDem}) \end{cases} \quad (8)$$

2.4. Cost objective

Cost, in euros, is calculated for the entire process of the work, from installation to waste treatment. During deconstruction phases, cost represent used resources such as workers and machines. In general, each resource is multiplied by the work duration (calculated result D) and the daily cost (DC in equations). Mechanical resources request additional cost for machines shipping from the company to the site. This last variable is called $Ship$ in the following equations.

Installation phase cost includes human and equipment resources to prepare the site (Eq. (9)). It could be bailiff statements, road protections, fence to close the plant, rent of a room for workers…

$$C_I = D_I \times (DC_{SF} \times SF_I + DC_W \times W_I) + Materials \quad (9)$$

Dismantling cost represents the cost of every resource with appropriate work duration (Eq. (10)). The site foreman is assumed to drive the front-end loader.

$$\begin{aligned} C_{Dis+LDis} = {} & MAX\,(D_{Dis}; D_{LDis}) \times DC_{SF} \times SF_{Dis} \\ & + D_{Dis1} \times [\,(DC_{ME} + DC_{MED}) \times ME_{Dis}] + D_{Dis} \times [DC_W \times W_{Dis}] \\ & + MAX\,(D_{Dis}; D_{LDis}) \times [DC_W \times W_{LDis} + DC_{FL} \times FL_{LDis}] \\ & + Ship_{ME} \times ME_{Dis} + Ship_{FL} \times \ FL_{LDis} \end{aligned} \quad (10)$$

The same approach is used for demolition cost (Eq. (11)). If demolition and loading phases are executed in the same time, then two excavators need to be transferred in the site, which comes down to two transfers ($Ship_E = 2$). For each excavator, one worker watches over the work from the ground. The site foreman is present only for the demolition phase, when risks are the most important.

$$\begin{cases} if\ Demol = 0 \Rightarrow C_{Dem+LDem} = D_{Dem} \times [DC_{SF} \times SF_{Dem} + DC_W \times W_{Dem} \\ \qquad\qquad + (DC_E + DC_{ED}) \times E_{Dem}\ ]+ \\ D_{LDem} \times [DC_W \times W_{LDem} + (DC_E + DC_{ED}) \times E_{LDem}] + Ship_E \times 1 \\ if\ Demol = 1 \Rightarrow C_{Dem+LDem} = D_{Dem} \times [DC_{SF} \times SF_{Dem} + DC_W \times W_{Dem} \\ \qquad\qquad + (DC_E + DC_{ED}) \times E_{Dem}\ ] \\ + MAX\,(D_{Dem}; D_{LDem}) \times [DC_{\mathrm{W}} \times W_{LDem} + (DC_E + DC_{ED}) \times E_{LDem}] \\ \qquad\qquad + Ship_E \times 2 \end{cases} \quad (11)$$

Next stage is waste management cost calculation, with transport ($C_{transport}$) and treatment ($C_{treatment}$). Transport cost represents waste evacuation with containers and trucks. The number of necessary containers depends on the choice of the volume or the mass that can contain a container. Containers are categorized into 3 types of waste: inert waste, wood and non-hazardous waste. Because of their diversity, wood and non-hazardous waste don't have the same density: this explains why volume are used in the calculation. This problem does not concern inert waste and transport calculations can be directly done with mass.

The following waste containers are used:

- For inert waste: 12 m^3, 15 m^3 with a simple container or 15 m^3 with a dumper truck
- For wood waste: 12 m^3, 30 m^3 or 60 m^3
- For non-hazardous waste, except wood: 12 m^3 or 30 m^3.

Containers filling are affected by two parameters:

- Expansion rate of waste (named ER with unit in %): it describes how waste can fill a larger volume than it really is, because of a non-optimized filling of the container
- Used volume or mass that can contain a container (named UV or UM with unit in m^3 or tons): it defines the mean filling level of a container.

Waste volume is multiplied by its expansion rate. Then the result is divided by the used volume of a container. The obtained number of containers is divided by the total number of daily truck turns that can be realized (*RD*). The result is finally multiplied by the daily cost of a truck (DC_{Truck}) to obtain transport cost (Eq. (12)). Cost is also influenced by the choice of waste treatment. In fact, if waste is evacuated to a landfill or a sorting plant, it is mixed in a same container, as long as it respects waste type (inert, non-hazardous and hazardous). Thus transport cost is saved.

$$\begin{aligned} C_{transport} = & \left[Rounded\ up\left(\frac{\sum M_{Inert}}{UM_{Inert}} \times \frac{1}{RD_{Inert}}\right) \times DC_{Truck} \right. \\ & + Rounded\ up\left(\frac{\sum V_{Non-hazardous} \times ER_{Non-hazardous} + V_{Wood} \times ER_{Wood}}{UV_{Non-hazardous}} \right. \\ & \left. \left. \times \frac{1}{RD_{Non-hazardous}}\right) \times DC_{Truck} \right]_{Landfill} \\ & + \left[Rounded\ up\left(\frac{\sum M_{Inert}}{UM_{Inert}} \times \frac{1}{RD_{Inert}}\right) \times DC_{Truck} \right. \\ & + Rounded\ up\left(\frac{\sum V_{Non-hazardous} \times ER_{Non-hazardous} + V_{Wood} \times ER_{Wood}}{UV_{Non-hazardous}} \right. \\ & \left. \left. \times \frac{1}{RD_{Non-hazardous}}\right) \times DC_{Truck} \right]_{Sorting} \\ & + \left[Rounded\ up\left(\sum \frac{M_{Inert}}{UM_{Inert}} \times \frac{1}{RD_{Inert}}\right) \times DC_{Truck} \right. \\ & + Rounded\ up\left(\sum \frac{V_{Non-hazardous} \times ER_{Non-hazardous}}{UV_{Non-hazardous}} \times \frac{1}{RD_{Non-hazardous}}\right) \\ & \left. \times DC_{Truck} + Rounded\ up\left(\sum \frac{V_{Wood} \times ER_{Wood}}{UV_{Wood}} \times \frac{1}{RD_{Wood}}\right) \times DC_{Truck} \right]_{Recycling} \end{aligned} \tag{12}$$

Waste treatment cost is calculated multiplying waste mass with corresponding treatment price, usually given per ton (Eq. (13)):

$$C_{Treatment} = \sum M_{Waste} \times TreatC_{Waste,treatment} \tag{13}$$

with M_{Waste} the waste mass (in tons) and $TreatC_{Waste,\ treatment}$ the treatment cost, that depends on the waste type and the treatment type (in euros per waste ton).

The cost objective aims to minimize sum of the different costs (Eq. (14)).

$$Minimize\ C = C_I + C_{Dis+LDis} + C_{Dem+LDis} + C_{transport} + C_{Treatment} \tag{14}$$

2.5. Recovering rate objective

The waste recovering rate - in % - enables to evaluate the respect of the European Union law [2]. Waste can be recovered through two schemes: either waste is sorted in a sorting plant and then is transported to a recovering plant or waste is directly sorted on site and is evacuated into a recovering plant. In a sorting plant, all recyclable waste cannot be sorted. It depends on the plant resources - manual or mechanical sorting, number of workers… - and the waste amount to manage. Then, for inert waste evacuated in a sorting plant, it is assumed that only 35% of the mass will reach a recovering plant. For non-hazardous waste, the recovering rate in a sorting plant is 15% [19]. For the second scheme, the recovering rate of waste is linked to the dismantling phase. With a correct dismantling, the recovering rate is assumed to be 100%. However, if dismantling is realized with an excavator during demolition, direct sorting is less efficient and the recovering rate is reduced. If dismantling is not realized, only some waste can be sorted in the site, because of their important volume, like concrete from building structure or wood from a roof. For other waste, direct transport to a recovering plant is impossible, because waste have not been sorted on the site. The recovering rates depend on each site: it must be calculated following identified waste in the building. Finally, if waste is evacuated for landfill, the recovering rate is 0%.

The recovering rate has to be maximized. However, in the multi-objective algorithm, the recovering rate is transformed in its negative set, which is landfill rate (Eq. (15)).

$$Minimize\ R = 1 - \frac{\sum R_{Dism,treat} \times M_{Waste}}{\sum M_{Waste}} \tag{15}$$

with $R_{Dism,\ treat}$ waste recovering rate, that depends on the variable *Dism* and the chosen treatment (*Treat* = 0, 1 or 2).

2.6. Constraint

In order to build strategies that can actually be implemented, different constraints have been set up:

- Recyclable waste that cannot be sorted without a correct dismantling is listed in this constraint, so that their recovering treatment will not be available if *Dism* = 1 or 2. For instance, impossible combinations for the case study are detailed in Table 5.
- If *Dism* = 1 or 2, human and mechanical resources, e.g. SF_{Dis} or W_{Dis}, for the dismantling phase and the loading of dismantling waste phase are zero.
- If *Dism* = 0, the number of site foremen for dismantling phase must at least respect the number given by Eq. (16). With this constraint, a strategy cannot propose a single person for the site, which is forbidden because of the risks.

$$SF_{Dis} = rounded.\ up\left(\frac{W_{Dis} + MED_{Dis} + W_{LDis}}{10}\right) \tag{16}$$

3. Optimization method

3.1. The problem implementation

With the 17 decision variables and their range (Table 2), more than 55 billion strategies can be built. A complete exploration of the solution space is not possible because of the whole set of equations and required data. To answer the combinatorial issue, metaheuristic (multi-objective search algorithms) is used to search optimal solutions. In the multi-objective algorithm, optimal solutions represent all non-dominated solutions in the Pareto meaning, i.e. a solution "x* is Pareto optimal if there exists no feasible vector x which would decrease some criterion without causing a simultaneous increase in at least one other criterion" [20].

In order to process optimization, the algorithm was implemented on JAVA using the MOEA Framework JAVA library by D. Hadka [21]. This library proposes a lot of search algorithms for multi-objective optimization. Some of these algorithms do not allow to define several constraints and only one is allowed. In order to be able to use them, constraints exposed in Section 2.6 have been aggregated into one.

3.2. Selection of algorithm

The choice of the search algorithm is a classic issue in optimization problem. There are numerous multi-objective search algorithms proposed in the literature. However, according to the 'No free lunch theorem' [22], no method is the best for all optimization contexts. To assess interest of the different algorithms regarding the deconstruction problem, eight of them were tested: NSGA-II, OMOPSO, CMA-ES, PAES, SPEA2, DBEA, MOEA/D and VEGA.

The *Non dominated Sorting Genetic Algorithm* (NSGA), in its second version (NSGA-II), is one of the most used algorithms. From a random population, solutions are sorted following number of other points that every solution dominates. Only non-dominated solutions are kept to create an offspring population. A new approach is then used on the next population. Parent and offspring are mixed in a single set. The new population is sorted with non-dominance relation. Then selection, crossover and mutation are applied on these non-dominated solutions.

For selection, crowding distance is used, assuring diversity of offspring. This process is iterated until the calculation ends [11]. NSGA-II has proven its efficiency for planning problems [23,24].

The OMOPSO algorithm is derived from *Particle Swarm Optimizer* (PSO). In this algorithm, solutions are seen as particles in the decision space. In each iteration, particles move according to a velocity vector. The velocity vector is defined by numerous factors like the best value of every single particle (pBest) and the value of one of the best particles among the population. These leader particles are sorted with a crowding factor, from population into an archive. Particles mutate when moving. A new population is born. The leader archive is updated. New leaders are identified among offspring if their actual value is better than the recorded pBest. If the archive contains too much leaders, the less efficient leaders are thrown away. Thus, with the new archive, the algorithm can move particles again and repeat the process [12].

The *Covariance Matrix Adaptation Evolution Strategy* (CMA-ES) algorithm is initially for single objective problems, but has been adapted for multiobjective optimization. The process begins with a population where each parent brings one offspring. Old and new generations are ranked with two criteria: level of non-dominance and contributing hypervolume. Best offspring update the covariance matrix, then are used as future parents for the next iteration. Some parameters are used to make the population evolve, such as the limited number of offspring and adaptation rate of the covariance matrix [13].

The *Pareto Archived Evolution Strategy* (PAES) algorithm is based on local search and minimal evolutions of the solutions with only use of mutation. For each iteration, one solution is selected to mutate. The parent and the offspring are compared. If the parent dominates, the process comes back to the beginning and chooses another solution. However, if the offspring dominates the parent, the offspring is then compared with non-dominated solutions contained in an archive. The offspring is kept in the archive if it is still non-dominated. Then the iteration is repeated [14]. The PAES algorithm is frequently used as a reference in algorithm comparisons [25].

In the second version of *Strength Pareto Evolutionary Algorithm* (SPEA 2), solutions are separated in two sets: the initial population or the archive, where non-dominated solutions are gathered. For each solution, the algorithm calculates its fitness value, which is the number of individuals that the solution dominates, and distance between the same solution and the rest of individuals. Solutions are non-dominated if their fitness value is lower than one. These non-dominated solutions become the new archive and are used with other selected solutions from the population to create a new generation of solutions. Then the algorithm repeats the cycle [15].

The *Decomposition-Based Evolutionary Algorithm* (DBEA) begins with a sampling of the population, ruled by reference points. To evaluate the population, distance is calculated between these reference vectors and solutions. Then pairs of solutions, chosen among the neighborhood or randomly, give birth to a new population. Children replace older solutions if they dominate them. Reference points are updated and sampled solutions are evaluated again [16].

The MOEA/D algorithm is a multi-objective evolutionary algorithm that transforms the multiobjective problem into numerous single objective problems. Each problem obtains an appropriate weight vector. With this, optimal solutions from each single problem must approach optimal solutions of the real multiobjective problem. In the framework of D. Hadka [21], the algorithm is a variant that uses differential evolution (MOEA/D-DE). Differential evolution is used to produce new solutions while keeping population diversity [17].

The *Vector Evaluated Genetic Algorithm* (VEGA) is one of the first created multiobjective algorithms. It distinguishes from other methods by separating randomly population in sub-populations. Each objective obtains a own sub-population and solutions of each sub-population are evaluated with their affiliated objective. Then VEGA follows the common process of multiobjective algorithms by creating a new generation, updating the actual population with the offspring and repeating the cycle [18]. It was successfully used for groundwater pollution problems looking into cost and reliability of the decontamination methods [26].

3.3. Criteria to compare algorithm

In order to define which of these search algorithms is the most relevant for the demolition strategy problem, four criteria are used: number of optimal solutions, variety of solutions regarding decision variables, percentage of solutions which respect the recovering regulation - 70% recovering rate -, and computing duration. For the first criterion - number of optimal solution -, we remove from the set of solutions those which are dominated by solutions obtained by another algorithm. The more an algorithm finds a high number of optimal solutions, the more the algorithm is considered as suitable. The second criterion - variety of solution - is computed by the normalized sum of standard deviation regarding decision variables. We assume that the more the algorithm provides various solutions, the more it offers possibilities for the decision-maker. The third criterion is a simple ratio between the number of solutions which have at least 70% of recovering rate and the total number of optimal solutions given by the algorithm. One objective of this work is to provide to the decision-maker interesting solutions in terms of cost and delay and also in agreement with the recovering rate regulation. The last criterion is obtained by measuring computation duration on an identical computer. The used computer, of the Fujitsu brand, has a processor Intel® Core™ i5-62000 CPU of 2,30 GHz.

4. Application

4.1. Case study presentation

In order to test the algorithm, a case study has been chosen among studies of the BDS company. The case is total deconstruction of seven neighboring buildings in a town center. Three buildings, with stone walls and a wooden structural frame, hosted shops on the first floor and apartments on the second floor with a floor structure in wood. The fourth building is constructed on a single floor, with concrete structure, wooden structural frame and metal facing. The fifth building hosted a bank and a lot of furniture have been left. The structure is in concrete on three floors, with a wooden structural frame. The sixth building hosted two shops on the first floor, then apartments on the two following floors. The structure is in concrete, as same as the roof. The last building is a single house on one level, with a stone structure and a wooden structural frame. In total, ground surface is 1070 m^2 and developed surface is 1943 m^2. Length of necessary fence is 100 m. Table 4 describes type, volume and mass of identified waste for this case.

For this case, feasible treatments and associated recovering rates are described in Table 5.

4.2. Strategy suggested by the company engineer

The case study is a real project that the company engineer studied with his experience and point of view. He proposed a deconstruction plan of 29 work days with a cost of 89 k€. The decision variables *Dism* and *Demol* are chosen as 0 - i.e. dismantling phase is correctly done and demolition and loading phases are done successively. Table 6 describes planned human and mechanical resources.

Table 7 defines the waste management plan proposed by the company engineer.

With this plan, the recovering rate reaches 80% of waste mass. Environmental objective is already obtained. Despite this performance, it is interesting to study if works can still improve in an environmental matter and if this criterion can be reached with smaller cost or duration.

4.3. Results

Each algorithm turned for approximatively 10 min with the following parameters:

- Iteration, i.e. number of optimization reprocesses: 5
- Evaluations: 50,000
- Population: 50.

The Table 8 gives the number of solutions and the minimum values for each algorithm on each objective.

Fig. 1 proposes a representation of the non-dominated solutions on a 3D-graph.

Fig. 2 illustrates optimal solutions distribution on the three pairs of objectives. Fig. 2-a and -b respectively represent Cost-Delay and Cost-Non-recovering rate pairs. Fig. 2-c illustrates Non-recovering rate-Delay pair.

In Fig. 2, optimal solutions of NSGA-II, OMOPSO, SPEA2 and CMA-ES are often masked by DBEA points. It means that all of these algorithms give similar solutions regarding objective performance. The Section 4.4.2 is dedicated to the discussion on the different algorithms. Number and quality of solutions will be analyzed.

4.4. Discussion

The aim of this study was twofold: (1) demonstrate capacity of the software to provide robust results, and if possible, better that the solution found by the company engineer, and (2) test different algorithms to select the most efficient. These two points will be discussed in the following parts.

4.4.1. Discussion on results

In this section, we will discuss the solutions given by the algorithms. Then we will study capacity of our algorithm to provide good results, and perhaps better results than the solution prepared by the company engineer. Among the whole set of solutions containing 290 solutions, 100 are really optimal. The 190 other solutions are dominated by at least one solution given by another algorithm. Cost goes from 65 k€ to 87 k€, with a delay work between 9 and 21 days and granted by a recovering rate from 46% to 81% - non-recovering rate from 54% to 19%.

Among the 100 optimal solutions, 83 solutions respect the environmental objective of 70% minimal recovering - a non-recovering rate equal or lower than 30% -, i.e. in agreement with the European Union objective. This environmental performance is obtained thanks to a dismantling process (decision variable *Dism* as 0) and a small use of landfills. In average, these solutions last 16 days for 80 k€ and a recovering rate of 76% (24% of non-recovering rate). In majority, 9 workers (W_{Dis}) and 2 mini-excavators (ME_{Dis}) are programmed for dismantling, then 3 workers for dismantling waste loading (ME_{Dis}). Programming two excavators at a same time (*Demol* as 1) reduces delays to 15 days in average, but barely does not affect cost. Among these solutions, 23 solutions obtained the best possible recovering rate, which is for this study case about 81%, meaning 19% of non-recovering rate (Fig. 2-c). For each waste, the best treatment is always chosen. It represents a mean cost of 82 k€ for 16 days of work. As before, 9 workers (W_{Dis}), 2 mini-excavators (ME_{Dis}) and 3 loading workers (W_{LDis}) are often programmed for dismantling.

By contrast, 17 optimal solutions do not reach the 70% recovering objective - See Fig. 2-b and -c. They mostly present the lowest cost and delay - less than 70 k€ and 15 days of work. These solutions correspond to a conventional demolition, without a dismantling phase (*Dism* as 1). Obviously, they have a lower non-recovering rate, which is about 50%.

Unsurprisingly, most of the solutions choose the biggest containers for waste. The dumper truck for inert waste evacuation has been selected only by one solution, while the 15 m^3 container was preferred because of its lower cost. None of the solutions proposes the variable *Dism* as 2, i.e. a little dismantling done by excavators when demolishing the building. This process represents too high cost and duration to be considered regarding the application case.

The algorithm has several benefits: 83% of the optimal solutions respect the environmental objective set by the European Union. Moreover, the algorithm has been able to combine expense and environmental concern. In these environmental solutions, 39 have a cost lower than the average of 80 k€, with 16 days of work in average. Among the solutions of 81% recovering rate, 12 of them propose a lower cost than the mean value - 81 k€ instead of 82 k€ -, but need in average two additional days - 18 instead of 16 days. In comparison with the partner plan - i.e. the plan which was really performed for this case study -, 17 solutions give the same recovering-rate of 80%, but with 7 k

Table 4
Description of identified waste in the case study.

Waste type	Waste	Volume (in m^3)	Mass (in tons)
Inert waste	Concrete	987.3	2172.0
	Glass	37.0	14.8
	Mixed inert waste	420.4	991.2
Non-hazardous waste	Wood	250.0	125.0
	Metal	158.3	19.0
	Plaster	69.2	90.0
	Plaster bricks	41.1	30.0
	Furniture	100.0	7.0
	Mixed non-hazardous waste	86.7	26.0
Total according to waste type	Inert waste	1444.7	3178.0
	Non-hazardous waste	705.3	297.0
	Inert and non-hazardous waste	2150.0	3475.0
Total according to deconstruction phase	Dismantling waste	542.3	211.8
	Demolition waste	1607.7	3263.2

Table 5
Recovering rate according to waste type and dismantling process (*Dism*).

Waste type	Treatment plant	*Dism* = 0	*Dism* = 1	*Dism* = 2
Concrete	Landfill	0%	0%	0%
	Sorting plant	35%	35%	35%
	Recovering plant	100%	70%	80%
Glass	Landfill	0%	0%	0%
	Sorting plant	35%	35%	35%
	Recovering plant	100%	Impossible	Impossible
Mixed inert waste	Landfill	0%	0%	0%
	Sorting plant	35%	35%	35%
	Recovering plant	Impossible	Impossible	Impossible
Wood	Landfill	0%	0%	0%
	Sorting plant	15%	15%	15%
	Recovering plant	100%	43%	75%
Metal	Landfill	0%	0%	0%
	Sorting plant	15%	15%	15%
	Recovering plant	100%	23%	45%
Plaster	Landfill	0%	0%	0%
	Sorting plant	15%	15%	15%
	Recovering plant	100%	Impossible	Impossible
Plaster bricks	Landfill	0%	0%	0%
	Sorting plant	15%	15%	15%
	Recovering plant	100%	Impossible	Impossible
Furniture	Landfill	0%	0%	0%
	Sorting plant	15%	15%	15%
	Recovering plant	100%	Impossible	Impossible
Mixed non-hazardous waste	Landfill	0%	0%	0%
	Sorting plant	15%	15%	15%
	Recovering plant	Impossible	Impossible	Impossible

Table 6
Type and number of planned human and mechanical resources for the case study.

	Human resources		Mechanical resources	
Deconstruction phase	Type	Number	Type	Number
Installation (*I*)	Site foreman (SF_I)	$(SF_I) = 1$		
	Worker (W_I)	$(W_I) = 2$		
Dismantling (*Dis*)	Site foreman (SF_{Dis})	$SF_{Dis} = 1$		
	Worker (W_{Dis})	$W_{Dis} = 3$		
	Mini-excavator driver (MED_{Dis})	$MSD_{Dis} = 1$	Mini-excavator (ME_{Dis})	$ME_{Dis} = 1$
Loading of dismantling waste (*LDis*)	Worker (W_{LDis})	$W_{LDis} = 1$	Front-end loader driven by the site foreman (FL_{LDis})	$FL_{LDis} = 1$
Demolition (*Dem*)	Site foreman (SF_{Dem})	$SF_{Dem} = 1$		
	Worker (W_{Dem})	$W_{Dem} = 1$		
	Excavator driver (ED_{Dem})	$ED_{Dem} = 1$	Excavator (E_{Dem})	$E_{Dem} = 1$
Loading of demolition waste (*LDem*)	Worker (W_{LDem})	$W_{LDem} = 1$		
	Excavator driver (ED_{LDem})	$ED_{LDem} = 1$	Excavator ($E_{L_{Dem}}$)	$E_{LDem} = 1$

Table 7
Waste treatments and containers planned for the case study.

Waste type	Waste	Treatment plant	Container
Inert waste	Concrete	Recovering plant	15 m^3
	Glass	Sorting plant	15 m^3
	Mixed inert	Sorting plant	15 m^3
Non-hazardous waste	Wood	Recovering plant	30 m^3
	Metal	Recovering plant	30 m^3
	Plaster	Recovering plant	30 m^3
	Plaster bricks	Landfill	30 m^3
	Furniture	Recovering plant	30 m^3
	Mixed non-hazardous	Sorting plant	30 m^3

Table 8
Set of optimal solutions according to the used algorithm for the case study.

Algorithm	Number of optimal solutions*	Min value for cost (in €)	Min value for delay (in days)	Min value for non-recovering rate (in %)
NSGA-II	66	64,755	9	19
OMOPSO	46	64,755	9	19
CMA-ES	40	76,141	13	19
PAES	24	75,787	13	19
SPEA2	44	75,101	13	19
DBEA	47	65,370	9	19
MOEA/D	22	75,410	13	20
VEGA	1	99,036	27	20
Total	290	64,755	9	19

* It is the direct number of solutions per algorithm, solutions dominated by other algorithm's solutions were not removed.

€ and 14 days less. Solutions with the 81% recovering rate are also better than the partner solution: 6 k€ and 13 days less. The algorithm has then succeeded to find more viable solutions in terms of cost, delay and environment.

4.4.2. Discussion on optimization algorithms

In this section, each algorithm will be analyzed in order to find the most relevant for the deconstruction problem. Table 9 gives algorithms performance for four criteria. Number of optimal solutions and computing duration are classic criteria to compare algorithms. The percentage of optimal solutions that reach at least 70% of recovering rate is interesting regarding the deconstruction problem in the European context. Variety of solutions can turn out to be interesting or not, depending on the optimization problem. Here, it was a demand from the partner company. The objective is to provide a set of solutions, optimal, but also sufficiently different to make the engineer think about the study and the various possible strategies.

Computing duration of the different algorithms is rather close; this criterion is not relevant to compare them. NSGA-II, SPEA2 and DBEA provide the higher number of optimal solutions whereas VEGA and MOEA/D, with respectively 0 and 1 solution, seem unsuitable to solve this problem. The failure of MOEA/D and VEGA can be due to three reasons: (i) these algorithms are not relevant according to our problem, (ii) default parameters are not adapted for our problem (In this work, we focused on several algorithms and not on the parameters changes for one single algorithm), (iii) they are not well implemented in the used library [25].

NSGA-II, SPEA2 and DBEA provide largest number of solutions. Their efficiencies are relatively close and it was not possible to decide which algorithm is the best with only this criterion. For solutions diversity, NSGA-II, OMOPSO and DBEA obtain the best results. Finally, PAES, CMA-ES and SPEA2 provide the highest percentage of solutions with a sufficient recovering rate to respect the regulation. The three algorithms NSGA-II, DBEA and SPEA2 appear as relevant for the deconstruction problem. However, it is difficult to assess which algorithm is the most efficient.

The company engineer that studied this deconstruction case has been asked to select the best set of solutions -each set per algorithm - between solutions given by NSGA-II, DBEA and SPEA2. For him, most relevant solutions are first those that obtain at least 70% of recovering rate, which means a necessary dismantling phase ($Dism = 0$). Human and mechanical resources need to be highly accurate in the case. In his opinion, for this deconstruction, two excavators in the same time is impossible - which means impossibility for $Demol = 1$ -, workers and mini-excavators for the dismantling phase (W_{Dis} and ME_{Dis}) cannot be more than respectively 6 and 2. Indeed, in this case study, several buildings with different access and surfaces need to be deconstructed, which tends to complicates movement of the mini-excavators and supervision by the site foreman.

With these new values, we turned again the algorithm for NSGA-II, DBEA and SPEA2. NSGA-II and SPEA2 propose solutions with the maximal capacity of resources, which means 6 W_{Dis} and 2 ME_{Dis} for 21 days of work and 80 k€, but little changes in the choice of waste treatment. By contrast, DBEA gives more diversity for resources, giving also strategies of 6 W_{Dis} and 1 ME_{Dis}, 5 W_{Dis} and 2 ME_{Dis} and 5 W_{Dis} and 0 ME_{Dis}. For the majority of the solutions, W_{LDis} are planned as 2. It implies an average cost of 84 k€ and an average delay of 24 days. In terms of cost and delays, strategies of these three algorithms usually dominate the company plan - i.e. engineer solution. Their environmental performance turns around a recovering rate of 76% - 24% of

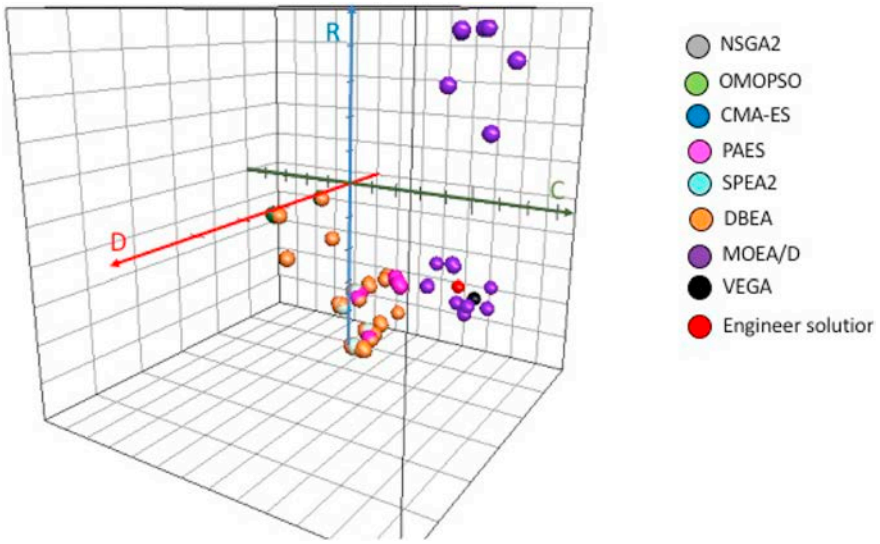


Fig. 1. Optimal solutions according to the used algorithm in the three objective axes - C for cost, D for delay and R for non-recovering rate.

non-recovering rate - and 46% of them obtain a recovering rate equal or higher than the company plan.

Due to the higher diversity of the solutions given by DBEA for this case-study, the company engineer preferred this algorithm. Indeed, from his point of view, it is better to obtain solutions with different work teams in order to select the one he judges the more relevant regarding the specificity of the case study project. Between DBEA solutions, he showed his interest for the strategies of 5 W_{Dis}, 2 ME_{Dis} and 2 W_{LDis}. With 22 days of work and a cost of 81 k€ to obtain a recovering rate of 80%, they could offer an interesting alternative for the initial plan.

In this experimentation, to assess the algorithm performance, choice was made to follow expert feedback rather than objective approach for several reasons: (i) objective approach does not really exist because the construction of a criterion, e.g. recycling/ratio, is yet a subjective choice; (ii) it is difficult to justify the choice of one criterion and its associated scale: for instance, creating a non-dimension scale assumes a normalization which could be criticized and finally, (iii) the study engineer wanted to be associated to the algorithm choice. However, it would be interesting to use a mixed approach: the engineer defines what would be for him a perfect solution regarding the case study and the distance between Pareto optimal solutions for each algorithm and this perfect solution is computed. These distances could be meaningful to decide between the algorithms. In addition, it would be interesting to make another optimization on new case studies to measure the robustness of the algorithm choice.

5. Conclusion

An innovative approach for optimizing building deconstruction strategy, which is a complex engineering decision problem, was suggested. Deconstruction planning was formulated as a Multi-Objective Optimization Problem to integrate different options regarding equipment, labors and waste treatment - decision variables - and to assess their interest in a global way, while taking into account the interrelations between these options. A real-case study, operated by the partner demolition company, was used to estimate the interest of the algorithm and its capacity to deal with real demolition project studies. The found optimized solutions were compared to the engineer solution. We choose to start the algorithm without taking into account this solution in order to measure the search algorithms capacity to find good solutions by themselves. Different search algorithms were tested, in order to define the most efficient algorithm for this kind of problem.

For the case-study, 83 optimal solutions propose a suitable relation between cost, delay and environment. They are all characterized by a dismantling process to lead waste to the best recovering, still regarding an acceptable cost. Work team composition is balanced between human and mechanical resources to reduce delays and cost as much as possible. The objective was completely reached: all of these optimal solutions outrank or equal the strategy proposed by the partner engineer. Cost and delay can be optimized even when an environmental performance is required. Among all tested algorithms, three algorithms - NGSA-II, DBEA and SPEA2 - appear as efficient for deconstruction and waste management planning problems. Between these three algorithms, DBEA proposed the set of preferred solutions by the partner company expert. Among these optimal solutions, one was determined as an interesting alternative to the engineer solution.

The algorithm studied in this paper was a prototype and still needs to be completed with more waste and machines. Nevertheless, the proposed algorithm already allows engineers to optimize deconstruction plan. Moreover, optimized plans were found in a reasonable time (less than 12 min) with a standard computer - Intel® Core™ i5-62000 CPU of 2,30 GHz. It ensures the capacity of the algorithm to be used in a real context and to give interesting results. The deconstruction company BDS will soon integrates its use in their project studies. The optimization process will propose two options: (1) if the engineer would like to gain time, directly use the algorithm to find optimized solutions, and (2) if he prefers to keep the control on the resulting plan, letting him propose a starting solution and begin the optimization by including it. The second option should lead to better results (or at least closer to the engineer point of view), but requires more involvement from her/him.

The algorithm was created so that it could be used in a different study context or in another demolition company by changing knowledge and case-study variables. To support this aspect, the algorithm will be presented to several French National organisms which are interested in building deconstruction (ADEME, CSTB…). One last perspective would be the connection with BIM in order to obtain the input data - amount of each waste, living area… - directly from the building algorithm, as proposed by Akbarnezhad et al. (2014) [27].

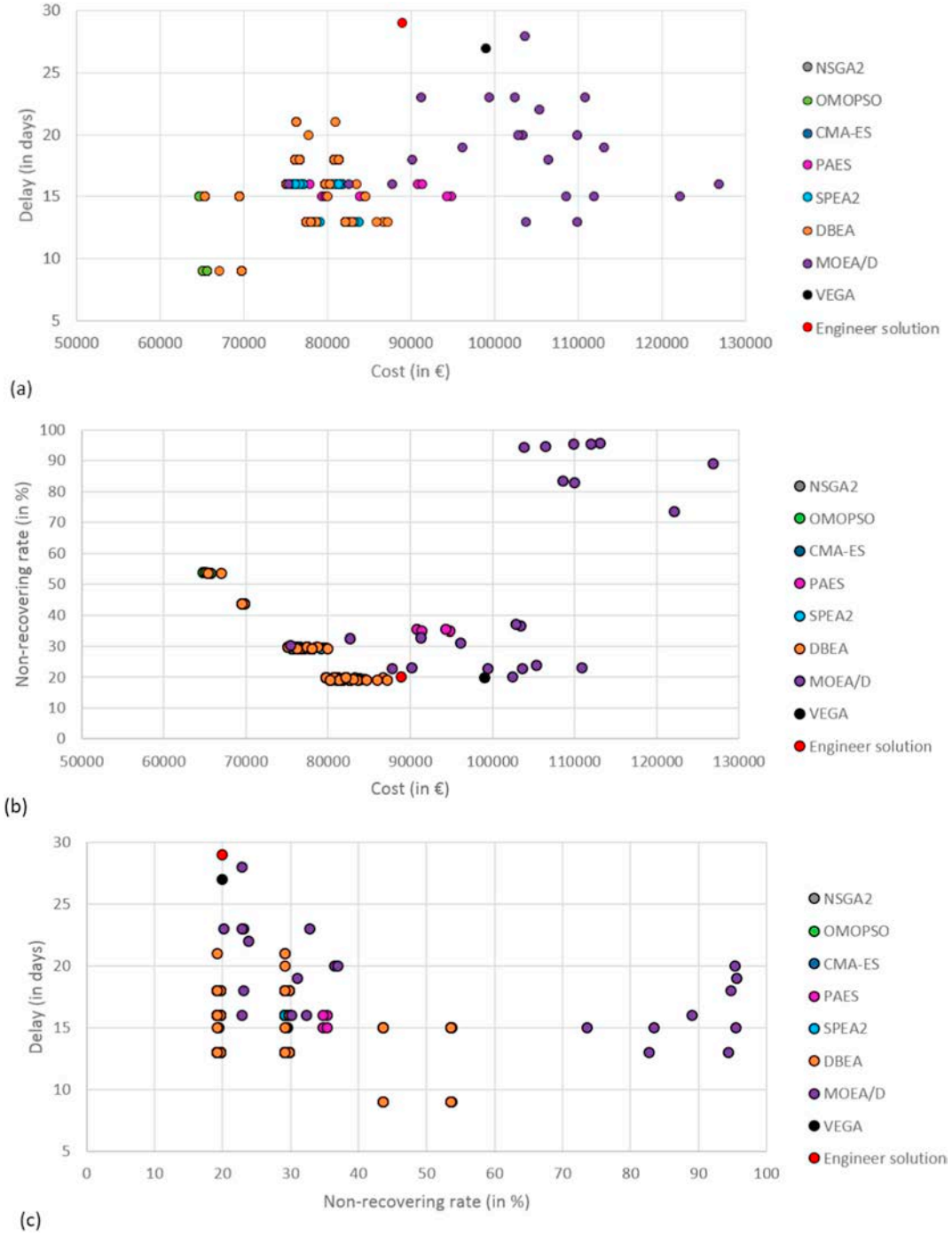


Fig. 2. Non-dominated solutions distribution in the three objective spaces - a for cost-delay, b for cost-non-recovering rate and c for non-recovering rate-delay.

Table 9
Search algorithm comparison for the case study.

Algorithm	Number of optimal solutions*	Variety of solutions	Computing duration (in min)	Percentage of optimal solutions that reach at least 70% of recovering rate (or less than 30% of non-recovering rate)
NSGA-II	21	0.55	10	76%
OMOPSO	14	0.58	11	57%
CMA-ES	6	0.08	12	100%
PAES	10	0.33	11	100%
SPEA2	23	0.27	12	100%
DBEA	25	0.53	12	84%
MOEA/D	1	–	12	0%
VEGA	0	–	11	–

* Here are considered only solutions which are optimal regarding the whole set of solutions.

Acknowledgment

The authors thank the deconstruction company BDS and its staff for providing the necessary information and opinion on the algorithms results.

References

[1] J. Koesten, Grands Principes de la Réglementation Relative à la Gestion des Déchets du BTP (Great Principles of the Regulations of Building Waste Management), Journée Technique des Déchets du BTP, CEREMA, 2015, http://www.nord-picardie.cerema.fr/IMG/pdf/CEREMA_Dter_NP_JT_D_R_chets_BTP_26112015_R_R_glementation_d_R_chets_BTP_cle61acfd.pdf , Accessed date: 13 July 2016.

[2] Parlement Européen et Conseil, Directive 2008/98/CE relative aux déchets (Directive 2008/98/EC on waste), http://www.ineris.fr/aida/consultation_document/35787, (2008) , Accessed date: 13 July 2016.

[3] Ecorys, EU construction & demolition waste management protocol, https://ec.europa.eu/growth/content/eu-construction-and-demolition-waste-protocol-0_en, (2016) , Accessed date: 22 December 2016.

[4] A. Dumont, A. Jacquet, C.-E. Bourge, P. Gosselin, B. Ingelaere, C. Legrand, C. Mertens, MéDéCo Métré Déchets Construction: Manuel de L'utilisateur (MeDeCo: User Guide), Office wallon des déchets, 2004, http://www.marco-construction.be , Accessed date: 19 July 2016.

[5] SNED, Tutoriel ivestigo: votre logiciel de traçabilité des déchets (Ivestigo guide: your software for waste traceability), http://www.sned.fr/mediatheque/Tutoriel_Ivestigo_v3.pdf, (2015) , Accessed date: 8 December 2016.

[6] Y. Li, X. Zhang, Web-based construction waste estimation system for building construction projects, Autom. Constr. 35 (2013) 142–156, https://doi.org/10.1016/j.autcon.2013.05.002.

[7] G. Banias, C. Achillas, C. Vlachokostas, N. Moussiopoulos, I. Papaioannou, A web-based Decision Support System for the optimal management of construction and demolition waste, Waste Manag. 31 (2011) 2497–2502, https://doi.org/10.1016/j.wasman.2011.07.018.

[8] T. Buhler, C. Buhe, C. Berdier, J.-F. Boscato, Pertinence et Faisabilité d'une Application d'aide à la Planification/Gestion de Déchets de Chantier: Du Chantier à la Gestion Territoriale (Relevance and feasibility of an aid application for plan/management of site waste), 25èmes Rencontres AUGC, Bordeaux, France, 2007, pp. 1–8 http://www.iut.u-bordeaux.fr/gc/augc07/index/pdf/ECE/Buhler.pdf , Accessed date: 25 May 2016.

[9] FFB, FNTP, Déchets et excédents de chantier, déchets chantier (Waste and site exceed, site waste), http://www.dechets-chantier.ffbatiment.fr/rechercher-centres.aspx, (2016) , Accessed date: 17 March 2017.

[10] F. Schultmann, B. Pitzini-Duée, T. Zundel, O. Rentz, Développement d'un logiciel d'audit de bâtiment avant démolition (Development of a software for building audit before demolition), Annales du Bâtiment et des Travaux Publics, 1998 2-86911-692-6, pp. 41–50.

[11] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature, Paris, France, September 18–20, 2000, pp. 849–858 (ISBN: 3-540-41056).

[12] M.R. Sierra, C. Coello Coello, Improving PSO-based multi-objective optimization using crowding, mutation and ϵ-dominance, Proceedings of the Third International Conference of Evolutionary Multi-Criterion Optimization, Guanajuato, Mexico, March 9–11, 2005 3-540-24983-4, pp. 505–519.

[13] C. Igel, N. Hansen, S. Roth, Covariance matrix adaptation for multi-objective optimization, Evol. Comput. 15 (2007) 1–28, https://doi.org/10.1162/evco.2007.15.1.1.

[14] J. Knowles, D. Corne, The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimization, Proceedings of the Congress of Evolutionary Computation, Washington, USA, July 6–9, 1999, pp. 1–8, , https://doi.org/10.1109/CEC.1999.781913.

[15] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the Strength Pareto Evolutionary Algorithm, TIK Report, (2001), pp. 1–22, https://doi.org/10.3929/ethz-a-004284029.

[16] M. Asafuddoula, T. Ray, R. Sarker, Decomposition based evolutionary algorithm for many objective optimization with systematic sampling and adaptive epsilon control, Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization, Sheffield, UK, March 19–22, 2013 978-3-642-37139-4, pp. 413–427, , https://doi.org/10.1007/978-3-37140-0.

[17] H. Li, Q. Zhang, Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, IEEE Trans. Evol. Comput. 13 (2007) 1–19, https://doi.org/10.1109/TEVC.2008.925798.

[18] T. El-Ghazali, Metaheuristics: From Design to Implementation, John Wiley & Sons, 978-0-470-27858-1, 2009.

[19] ADEME, Treize Développement, Pöyry SAS, Analyse technico-économique de 39 plateformes françaises de tri/valorisation des déchets du BTP (Technical-economic analysis of 39 French centers of sorting/recycling building waste) (Synthèse réalisée pour l'ADEME), http://www.ademe.fr/analyse-technico-economique-39-plate-formes-francaises-trivalorisation-dechets-btp, (2011) , Accessed date: 30 September 2016.

[20] C. Coello Coello, G. Lamont, D. Van Veldhuizen, Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd edition, Springer, 978-0-387-33254-3, 2007.

[21] D. Hadka, MOEA framework, http://moeaframework.org/index.html, (2017) , Accessed date: 20 July 2017.

[22] D.H. Wolpert, W.G. MacReady, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. (1997) 67–82, https://doi.org/10.1109/4235.585893.

[23] P. Murugan, S. Kannan, S. Baskar, NSGA-II algorithm for multi-objective generation expansion planning problem, Electr. Power Syst. Res. 79 (2009) 622–628, https://doi.org/10.1016/j.epsr.2008.09.011.

[24] M.S. El-Abbasy, A. Elazouni, T. Zayed, MOSCOPEA: multi-objective construction scheduling optimization using elitist non-dominated sorting genetic algorithm, Autom. Constr. 71 ((2016) 153–170, https://doi.org/10.1016/j.autcon.2016.08.038.

[25] D. Hadka, Beginner's Guide to the MOEA Framework (for Version 2.12 and Later), (2016) 978-1-329-82596-3.

[26] B.J. Ritzel, J.W. Eheart, S. Ranjithan, Using genetic algorithms to solve a multiple objective groundwater pollution containment problem, Water Resour. Res. 30 (1994) 1589–1603, https://doi.org/10.1029/93WR03511.

[27] A. Akbarnezhad, K.C. Ong, L.R. Chandra, Economic and environmental assessment of deconstruction strategies using building information algorithming, Autom. Constr. 37 (2014) 131–144, https://doi.org/10.1016/j.autcon.2013.10.017.