Dipartimento di Informatica
e Sistemistica Antonio Ruberti

Sapienza
Università di Roma

Feasibility Pump-Like Heuristics for Mixed Integer Problems

Marianna De Santis
Stefano Lucidi
Francesco Rinaldi

Technical Report n. 15, 2010

Feasibility Pump-Like Heuristics for Mixed Integer Problems

M. De Santis, S. Lucidi, F. Rinaldi

Dipartimento di Informatica e Sistemistica
Sapienza Università di Roma
Via Ariosto, 25 - 00185 Roma - Italy

e-mail: mdesantis@dis.uniroma1.it
e-mail: stefano.lucidi@dis.uniroma1.it
e-mail: rinaldi@dis.uniroma1.it

Abstract

Mixed-Integer optimization represents a powerful tool for modelling many optimization problems arising from real-world applications. The Feasibility pump is a heuristic for finding feasible solutions to mixed integer linear problems. In this work, we propose a new feasibility pump approach for MIP problems using concave non differentiable penalty functions for measuring solution integrality.

Keywords. Mixed integer programming, Concave penalty functions, Frank-Wolfe algorithm, Feasibility Pump.

MSC. 90C06, 90C10, 90C11, 90C30, 90C59

1 Introduction

Many real-world problems can be modeled as Mixed Integer Programming (MIP) problems, namely as minimization problems where some (or all) of the variables only assume integer values. Finding a first feasible solution quickly is crucial for solving this class of problems. In fact, many local-search approaches for MIP problems such as Local Branching [11], guide dives and RINS [8] can be used only if a feasible solution is available.

In the literature, several heuristics methods for finding a first feasible solution for a MIP problem have been proposed (see e.g. [2]-[4], [6], [13]-[18]). Recently, Fischetti, Glover and Lodi [10] proposed a new heuristic, the well-known Feasibility Pump, that turned out to be very useful in finding a first feasible solution even when dealing with hard MIP instances. The FP heuristic is implemented in various MIP solvers such as BONMIN [7].

The basic idea of the FP is that of generating two sequences of points $\{\bar{x}^k\}$ and $\{\tilde{x}^k\}$ such that $\bar{x}^k$ is LP-feasible, but may not be integer feasible, and $\tilde{x}^k$ is integer, but not necessarily LP-feasible. To be more specific the algorithm starts with a solution of the LP relaxation $\bar{x}^0$ and sets $\tilde{x}^0$ equal to the rounding of $\bar{x}^0$. Then, at each iteration $\bar{x}^{k+1}$ is chosen as the nearest LP-feasible point in ℓ_1-norm to $\tilde{x}^k$, and $\tilde{x}^{k+1}$ is obtained as the rounding of $\bar{x}^{k+1}$. The aim of the algorithm is to reduce at each iteration the distance between the points of the two sequences, until the two points are the same and an integer feasible solution is found. Unfortunately, it can happen that the distance between $\bar{x}^{k+1}$ and $\tilde{x}^k$ is greater than zero and $\tilde{x}^{k+1} = \tilde{x}^k$, and the strategy can stall. In order to overcome this drawback, random perturbations and restart procedures are performed.

As the algorithm has proved to be effective in practice, various papers devoted to its further improvements have been developed. Fischetti, Bertacco and Lodi [5] extended the ideas on which the FP is based in two different directions: handling MIP problems with both 0-1 and integer variables, and exploiting the FP information to drive a subsequent enumeration phase. In [1], in order to improve the quality of the feasible solution found, Achterberg and Berthold consider an alternative distance function which takes into account the original objective function. In [12], Fischetti and Salvagnin proposed a new rounding heuristic based on a diving-like procedure and constraint propagation.

An interesting interpretation of the FP has been given by J.Eckstein and M.Nediak in [6]. In this work they noticed that the FP heuristic may be seen as a form of Frank-Wolfe procedure applied to a nonsmooth merit function which penalizes the violation of the 0-1 constraints.

In this paper, we extend to the case of general MIP problems the approach described in [9] for finding a first feasible solution to binary MIP problems.

The paper is organized as follows. In Section 2, we give a brief review of the Feasibility Pump heuristic for general MIP problems. In Section 3, we show the equivalence between the FP heuristic and the Frank-Wolfe algorithm applied to a nonsmooth merit function. In Section 4, we introduce new nonsmooth merit functions for dealing with general integer variables. Finally, we present our algorithm in Section 5.

In the following, given a concave function $f : R^n \to R$, we denote by $\partial f(x)$ the set of supergradients of f at the point x, namely

$$\partial f(x) = \{v \in R^n \;:\; f(y) - f(x) \leq v^T(y-x),\; \forall\, y \in R^n\}.$$

2 The Feasibility Pump Heuristic for General MIP Problems

We consider a MIP problem of the form:

$$\begin{aligned} &\min c^T x \\ &\text{s.t. } Ax \geq b \\ &\quad l \leq x \leq u \\ &\quad x_j \in \mathbb{Z}\ \forall j \in I, \end{aligned} \tag{MIP}$$

where $A \in \mathbf{R}^{m \times n}$ and $I \subset \{1, 2, \ldots, n\}$ is the set of indices of integer variables. Let $P = \{x : Ax \geq b,\ l \leq x \leq u\}$ denote the polyhedron of the LP-relaxation of (MIP). The Feasibility Pump starts from the solution of the LP relaxation problem $\bar{x}^0 := \arg\min\{c^T x : x \in P\}$ and generates two sequences of points $\bar{x}^k$ and $\tilde{x}^k$: $\bar{x}^k$ is LP-feasible, but may be integer infeasible; $\tilde{x}^k$ is integer, but not necessarily LP-feasible. At each iteration $\bar{x}^{k+1} \in P$ is the nearest point in ℓ_1-norm to $\tilde{x}^k$:

$$\bar{x}^{k+1} := \arg\min_{x \in P} \Delta(x, \tilde{x}^k) \tag{1}$$

where

$$\Delta(x, \tilde{x}^k) = \sum_{j \in I} |x_j - \tilde{x}_j^k|.$$

It is easy to notice that solving problem (1) is equivalent to solve the following LP-problem:

$$\begin{aligned} \min \quad & \sum_{j \in I : \tilde{x}_j^k = l_j} (x_j - l_j) + \sum_{j \in I : \tilde{x}_j^k = u_j} (u_j - x_j) + \sum_{j \in I : l_j < \tilde{x}_j^k < u_j} d_j \\ \text{s.t.} \quad & Ax \geq b \\ & l \leq x \leq u \\ & -d_j \leq x_j - \tilde{x}_j^k \leq d_j \qquad \forall j \in I :\ l_j < \tilde{x}_j^k < u_j\ , \end{aligned} \tag{2}$$

where the variables d_j are introduced to model the nonlinear function $d_j = |x_j - \tilde{x}_j^k|$ for integer variables x_j that are not equal to one of their bounds in the rounded solution $\tilde{x}^k$.
The point $\tilde{x}^{k+1}$ is obtained as the rounding of $\bar{x}^{k+1}$. The procedure stops if at some iteration l, $\bar{x}^l$ is integer or, in case of failing, if it reaches a time or iteration limit. In order to avoid stalling issues and loops, the Feasibility Pump performs a perturbation step. Here we report a brief outline of the basic scheme:

The Feasibility Pump (FP) for general MIPs - basic version

Initialization: Set $k = 0$, let $\bar{x}^0 := \arg\min\{c^T x : x \in P\}$
If ($\bar{x}^0$ is integer) return $\bar{x}^0$
Compute $\tilde{x}^0 = round(\bar{x}^0)$
While (not stopping condition) **do**

Step 1 Compute $\bar{x}^{k+1} := \arg\min\{\Delta(x, \tilde{x}^k) : x \in P\}$

Step 2 If ($\bar{x}^{k+1}$ is integer) return $\bar{x}^{k+1}$

Step 3 Compute $\tilde{x}^{k+1} = round(\bar{x}^{k+1})$

Step 4 If (cycle detected) $\tilde{x}^{k+1} = perturb(\tilde{x}^k)$

Step 5 Update $k = k + 1$

End While

Now we give a better description of the rounding and the perturbing procedures used respectively at **Step 3** and at **Step 4**:

Round: This function transforms a given point $\bar{x}^k$ into an integer one, $\tilde{x}^k$. The easiest choice is that of rounding each component $\bar{x}^k_j$ with $j \in I$ to the nearest integer, while leaving the continuous components of the solution unchanged. Formally,

$$\tilde{x}^k_j = \begin{cases} [\bar{x}^k_j] = \lfloor \bar{x}^k_j + \tau \rfloor & \text{if } j \in I \\ \\ \bar{x}^k_j & \text{otherwise} \end{cases} \tag{3}$$

where $\tau = 0.5$, and $\lfloor \cdot \rfloor$ represents the floor function (a function which maps a real number to the largest previous integer). Another possibility is that of using a random τ like that described in [5]:

$$\tau(\omega) = \begin{cases} 2\omega(1-\omega), & \text{if } \omega \leq \dfrac{1}{2} \\ \\ 1 - 2\omega(1-\omega), & \text{otherwise} \end{cases} \tag{4}$$

where ω is a uniform random variable in $[0, 1)$. Using the definition (4), threshold τ can assume a value between 0 and 1, but values close to 0.5 are more likely than those close to 0 or 1.

Perturb: The aim of the perturbation procedure is to avoid cycling and it consists in two heuristics. To be more specific:

- if $\tilde{x}^k_j = \tilde{x}^{k+1}_j$ for all $j \in I$ a weak perturbation is performed, namely, a random number of integer constrained components, chosen as to minimize the increase in the distance $\Delta(\bar{x}^{k+1}, \tilde{x}^{k+1})$, is moved using the following rule:

$$\tilde{x}^{k+1}_j = \begin{cases} \tilde{x}^k_j + 1, & \text{if } \bar{x}^{k+1}_j > \tilde{x}^k_j \\ \\ \tilde{x}^k_j - 1, & \text{if } \bar{x}^{k+1}_j < \tilde{x}^k_j \end{cases} \tag{5}$$

– A *restart* operation, consisting of random perturbation of some entries of $\tilde{x}^{k+1}$, is performed if one of the following situations occur:
  - the point $\tilde{x}^{k+1}$ is equal, in its integer components, to a previously generated point;
  - the distance $\Delta(\bar{x}^{k+1}, \tilde{x}^{k+1})$ did not decrease by at least 10% in the last KK iterations.

3 The FP heuristic and the Frank-Wolfe method

In this section, following a similar reasoning as in [6, 9], we point out the equivalence between the FP heuristic and the Frank-Wolfe method for the general integer case. In order to better understand this equivalence we briefly recall the unitary stepsize Frank-Wolfe method for concave non-differentiable functions. Let us consider the problem

$$\begin{aligned} &\min f(x) \\ &x \in P \end{aligned} \tag{6}$$

where $P \subset R^n$ is a non empty polyhedral set that does not contain lines going to infinity in both directions, $f : R^n \to R$ is a concave, non-differentiable function, bounded below on P.
The Frank-Wolfe algorithm with unitary stepsize (see [9] for further details) at each iteration k produces a new point

$$x^{k+1} = \arg\min_{x \in P} \ (g^k)^T x$$

where $g^k \in \partial f(x^k)$. Then, the algorithm involves only the solution of linear programming problems, and it is proved in [20] that it converges to a stationary point $x^\star$ in a finite number of iterations.
Now we consider the basic FP heuristic without any perturbation (i.e. without Step 4) and we show that it can be interpreted as the Frank-Wolfe algorithm with unitary stepsize applied to a concave, nondifferentiable merit function.
First of all, we can rewrite the distance as follows:

$$\Delta(x, \tilde{x}^k) = \sum_{j \in I : \tilde{x}_j^k = l_j} x_j - \sum_{j \in I : \tilde{x}_j^k = u_j} x_j + \sum_{j \in I : l_j < \tilde{x}_j^k < u_j} d_j.$$

At each iteration, the Feasibility Pump for MILP problems with general integers computes, at Step 1, the solution of the LP problem (2), then, at Step 3, it rounds the solution $\bar{x}^k$, thus giving $\tilde{x}^{k+1}$.
These two operations can be included in the unique step of calculating the solution of the following LP problem:

$$\begin{aligned} \min \quad & \sum_{j \in I : \bar{x}_j^k \le l_j + \frac{1}{2}} x_j - \sum_{j \in I : \bar{x}_j^k > u_j - \frac{1}{2}} x_j + \sum_{j \in I : l_j + \frac{1}{2} < \bar{x}_j^k \le u_j - \frac{1}{2}} d_j \\ \text{s.t.} \quad & Ax \ge b \\ & l \le x \le u \\ & -d_j \le x_j - [\bar{x}_j^k] \le d_j \qquad \forall j \in I : \ l_j + \tfrac{1}{2} < \bar{x}_j^k \le u_j - \tfrac{1}{2} \ . \end{aligned} \tag{7}$$

It can be proved that problem (7) is equivalent to

$$\begin{aligned} \min \quad & \sum_{j \in I: \bar{x}_j^k \leq l_j + \frac{1}{2}} x_j - \sum_{j \in I: \bar{x}_j^k > u_j - \frac{1}{2}} x_j + \sum_{\substack{j \in I,\ i=1,\dots,m_j: \\ s_{ij} - \frac{1}{2} < \bar{x}_j^k \leq s_{ij} + \frac{1}{2}}} d_{ij} \\ \text{s.t.} \quad & Ax \geq b \\ & l \leq x \leq u \\ & -d_{ij} \leq x_j - s_{ij} \leq d_{ij} \qquad \forall j \in I,\ i = 1, \dots, m_j, \end{aligned} \tag{8}$$

with $\{s_{ij} :\ i = 1, \dots, m_j\} = (l_j, u_j) \cap Z$.

Proposition 1 *Problem (7) is equivalent to problem (8).*

Proof. We can easily notice that for each $j \in I$ there exists only one index $i_j \in \{1, \dots, m_j\}$ such that

$$s_{i_j j} = [\bar{x}_j^k],$$

then we can rewrite the objective function of problem (8) as follows

$$\sum_{j \in I: \bar{x}_j^k \leq l_j + \frac{1}{2}} x_j - \sum_{j \in I: \bar{x}_j^k > u_j - \frac{1}{2}} x_j + \sum_{j \in I: l_j + \frac{1}{2} < \bar{x}_j^k \leq u_j - \frac{1}{2}} d_{i_j j}.$$

The other variables d_{ij} can assume an arbitrarily large value, so that the related constraints are trivially satisfied. Then problem (8) becomes

$$\begin{aligned} \min \quad & \sum_{j \in I: \bar{x}_j^k \leq l_j + \frac{1}{2}} x_j - \sum_{j \in I: \bar{x}_j^k > u_j - \frac{1}{2}} x_j + \sum_{j \in I: l_j + \frac{1}{2} < \bar{x}_j^k \leq u_j - \frac{1}{2}} d_{i_j j} \\ \text{s.t.} \quad & Ax \geq b \\ & l \leq x \leq u \\ & -d_{i_j j} \leq x_j - [\bar{x}_j^k] \leq d_{i_j j} \qquad \forall j \in I :\ l_j + \tfrac{1}{2} < \bar{x}_j^k \leq u_j - \tfrac{1}{2}\ . \end{aligned} \tag{9}$$

By setting variables $d_{i_j j}$ equal to d_j in (9), we have problem (7). □

Problem (8) can be seen as the iteration of the Frank Wolfe method with unitary stepsize applied to the following minimization problem

$$\begin{aligned} \min \ & \sum_{j \in I} \min\{x_j - l_j, u_j - x_j, d_{1j}, \dots, d_{m_j j}\} \\ \text{s.t.} \ & Ax \geq b \\ & l \leq x \leq u \\ & -d_{ij} \leq x_j - s_{ij} \leq d_{ij} \qquad \forall j \in I,\ i = 1, \dots, m_j\ . \end{aligned} \tag{10}$$

4 Nonsmooth merit functions for solving general MIPs

In the previous section, we have seen that the Feasibility Pump for general MIP problems is equivalent to the Frank Wolfe algorithm applied to the following problem

$$
\begin{aligned}
\min\ \psi(x,d) = & \sum_{j\in I} \min\{\phi(x_j - l_j), \phi(u_j - x_j), \phi(d_{1j}), \ldots, \phi(d_{m_j j})\} \\
\text{s.t.}\ \ & Ax \geq b \\
& l \leq x \leq u \\
& -d_{ij} \leq x_j - s_{ij} \leq d_{ij} \qquad \forall j \in I,\ i = 1,\ldots,m_j\ .
\end{aligned}
\tag{11}
$$

where $d = (d_{11}, \ldots, d_{m_1 1}, d_{12}, \ldots, d_{m_2 2}, \ldots, d_{1N}, \ldots, d_{m_N N})$ and $N = |I|$.
The function $\phi : R \to R$ is the identity function

$$\phi(t) = t.$$

Following the idea in [9, 19], we replace the linear terms in ϕ with suitable nonlinear terms that lead to a merit function whose feature is that of encouraging the change of a bunch of variables rather than distributing this change over all the variables.

Here are the terms we use:

Logarithmic function

$$\phi(t) = \ln(t + \varepsilon) \tag{12}$$

Hyperbolic function

$$\phi(t) = -(t + \varepsilon)^{-p} \tag{13}$$

Concave function

$$\phi(t) = 1 - \exp(-\alpha t) \tag{14}$$

Logistic function

$$\phi(t) = [1 + \exp(-\alpha t)]^{-1} \tag{15}$$

where $\varepsilon, \alpha, p > 0$.

5 The Reweighted Feasibility Pump for MIPs

As in [9], the use of the ϕ functions (12)-(15) leads to a new FP scheme in which the ℓ_1-norm used for calculating the next LP-feasible point is replaced with a "weighted" ℓ_1-norm of the form

$$\Delta_W(x, \tilde{x}) = \sum_{j\in I:\tilde{x}_j^k = l_j} w_j(x_j - l_j) + \sum_{j\in I:\tilde{x}_j^k = u_j} w_j(u_j - x_j) + \sum_{j\in I: l_j < \tilde{x}_j^k < u_j} w_j d_j \tag{16}$$

where the variables $d_j = |x_j - \tilde{x}_j|$ satisfy the constraints

$$-d_j \leq x_j - \tilde{x}_j \leq d_j \qquad \forall j \in I :\ l_j < \tilde{x}_j < u_j, \tag{17}$$

and w_j, $j = 1, \ldots, n$ are positive weights depending on the ϕ term chosen. Here we report an outline of the algorithm:

Reweighted Feasibility Pump (RFP) for general MIPs - basic version

Initialization: Set $k = 0$, let $\bar{x}^0 := \arg\min\{c^T x : Ax \geq b\}$
If ($\bar{x}^0$ is integer) return $\bar{x}^0$
Compute $\tilde{x}^0 = round(\bar{x}^0)$
While (not stopping condition) **do**

Step 1 Compute $\bar{x}^{k+1} := \arg\min\{\Delta_{W^k}(x, \tilde{x}^k) : Ax \geq b\}$

Step 2 **If** ($\bar{x}^{k+1}$ is integer) return $\bar{x}^{k+1}$

Step 3 Compute $\tilde{x}^{k+1} = round(\bar{x}^{k+1})$

Step 4 **If** (cycle detected) $(\tilde{x}^{k+1}, \bar{x}^{k+1}) = perturb(\tilde{x}^k, \bar{x}^k)$

Step 5 Update $k = k + 1$

End While

We assume that the *round* procedure is the same as that described in Section 2 for the original version of the FP heuristic. As regards the *perturb* procedure, we first perturb the point $\tilde{x}^k$ using the same procedure as that described in Section 2, then for all indices $j \in I$ such that $\tilde{x}^{k+1} \neq \tilde{x}^k$, we add 0.5 to $\bar{x}_j^{k+1}$ (if $\bar{x}_j^{k+1} > \tilde{x}_j^k$) or subtract 0.5 to $\bar{x}_j^{k+1}$ (if $\bar{x}_j^{k+1} < \tilde{x}_j^k$). Anyway, different rounding and perturbing procedures can be suitably developed.
Following the same reasoning of Section 3, we can reinterpret the reweighted FP heuristic without perturbation as the unitary stepsize Frank-Wolfe algorithm applied to the merit function ψ. Let us now consider a generic iteration k of the reweighted FP. At Step 1, the algorithm computes the solution of the LP problem

$$\begin{aligned}
\min \quad & \sum_{j \in I: \bar{x}_j^k \leq l_j + \frac{1}{2}} w_j^k x_j - \sum_{j \in I: \bar{x}_j^k > u_j - \frac{1}{2}} w_j^k x_j + \sum_{j \in I: l_j + \frac{1}{2} < \bar{x}_j^k \leq u_j - \frac{1}{2}} w_j^k d_j \\
\text{s.t.} \quad & Ax \geq b \\
& l \leq x \leq u \\
& -d_j \leq x_j - [\bar{x}_j^k] \leq d_j \qquad \forall j \in I: \; l_j + \tfrac{1}{2} < \bar{x}_j^k \leq u_j - \tfrac{1}{2} \; .
\end{aligned} \tag{18}$$

As in Section 3, we can prove that problem (18) is equivalent to

$$\begin{aligned}
\min \quad & \sum_{j \in I: \bar{x}_j^k \leq l_j + \frac{1}{2}} \tilde{w}_j^k x_j - \sum_{j \in I: \bar{x}_j^k > u_j - \frac{1}{2}} \tilde{w}_j^k x_j + \sum_{\substack{j \in I, \; i=1,\ldots,m_j: \\ s_{ij} - \frac{1}{2} < \bar{x}_j^k \leq s_{ij} + \frac{1}{2}}} \tilde{w}_{ij}^k d_{ij} \\
\text{s.t.} \quad & Ax \geq b \\
& l \leq x \leq u \\
& -d_{ij} \leq x_j - s_{ij} \leq d_{ij} \qquad \forall j \in I, \; i = 1, \ldots, m_j,
\end{aligned} \tag{19}$$

with $\{s_{ij}:\ i=1,\dots,m_j\}=(l_j,u_j)\cap Z$.

By setting

$$\tilde{w}_j^k = |g^k_{x_j}|$$

$$\tilde{w}_{ij}^k = |g^k_{d_{ij}}|$$

with $g^k\in\partial\psi(\bar{x}^k,\bar{d}^k)$, and $\bar{d}^k_{ij}=|x_j-s_{ij}|$, $\forall j\in I,\ i=1,\dots,m_j$.
Problem (19) can be seen as the iteration of the Frank Wolfe method with unitary stepsize applied to the following minimization problem

$$\begin{aligned}
&\min \psi(x,d)=\sum_{j\in I}\min\{\phi(x_j-l_j),\phi(u_j-x_j),\phi(d_{1j}),\dots,\phi(d_{m_jj})\}\\
&\text{s.t.}\ \ Ax\geq b \\
&\qquad l\leq x\leq u \\
&\qquad -d_{ij}\leq x_j-s_{ij}\leq d_{ij} \qquad \forall j\in I,\ i=1,\dots,m_j\ .
\end{aligned} \tag{20}$$

In order to better understand the meaning of the Reweighted Feasibility Pump we give an example.

Example 1 *By choosing ϕ equal to the logarithmic function, we can write problem (20) as follows:*

$$\begin{aligned}
&\min \psi(x,d)=\sum_{j\in I}\min\{\log(x_j-l_j+\epsilon),\log(u_j-x_j+\epsilon),\log(d_{1j}+\epsilon),\dots,\log(d_{m_jj}+\epsilon)\}\\
&s.t.\ \ Ax\geq b \\
&\qquad l\leq x\leq u \\
&\qquad -d_{ij}\leq x_j-s_{ij}\leq d_{ij} \qquad \forall j\in I,\ i=1,\dots,m_j\ .
\end{aligned} \tag{21}$$

Then, at an iteration k of the Reweighted Feasibility Pump heuristic, we have

$$w_j^k=\begin{cases}
\tilde{w}_j^k=\dfrac{1}{\bar{x}_j^k-l_j+\epsilon} & \text{if }\ \bar{x}_j^k\leq l_j+1/2\\[2ex]
\tilde{w}_j^k=\dfrac{1}{u_j-\bar{x}_j^k+\epsilon} & \text{if }\ \bar{x}_j^k\geq u_j-1/2\\[2ex]
\tilde{w}_{ij}^k=\dfrac{1}{\bar{d}_{ij}^k+\epsilon} & \text{if }\ s_{ij}-1/2\leq \bar{x}_j^k\leq s_{ij}+1/2
\end{cases} \tag{22}$$

References

[1] T. Achterberg and T. Berthold. *Improving the feasibility pump.* Discrete Optimization, 4, pp 77–86, 2007.

[2] E. Balas, S. Ceria, M. Dawande, F. Margot, G. Pataki. *OCTANE: A new heuristic for pure 0-1 programs.* Operations Research, 49(2), pp 207-225, 2001.

[3] E. Balas, C.H. Martin. *Pivot-and-complement: A heuristic for 0-1 programming.* Management Science, 26(1), pp 86-96, 1980.

[4] E. Balas, S. Schmieta, C. Wallace.*Pivot and shifta mixed integer programming heuristic.* Discrete Optimization, 1(1), pp 3-12, 2004.

[5] L. Bertacco, M. Fischetti, and A. Lodi. *A feasibility pump heuristic for general mixed-integer problems.* Discrete Optimization, 4, pp 63–76, 2007.

[6] J. Eckstein and M. Nediak. *Pivot, cut, and dive: a heuristic for 0-1 mixed integer programming.* Journal of Heuristics, 13, pp 471–503, 2007.

[7] P. Bonami, L.T. Biegler, A.R. Conn, G. Cornuejols, I.E. Grossmann, C.D. Laird, J. Lee, A. Lodi, F. Margot, N.Sawaya and A. Waechter. *An Algorithmic Framework for Convex Mixed Integer Nonlinear Programs.* IBM Research Report RC23771, To appear in Discrete Optimization, in press.

[8] E. Danna, E. Rothberg, C. Le Pape. *Exploring relation induced neighborhoods to improve MIP solution.* Mathematical Programming 102, 1, pp 71–90, 2005.

[9] De Santis M., Lucidi S., Rinaldi F. *New concave penalty functions for improving the Feasibility Pump*, DIS Technical Report, Vol 2, n.10 (2010).

[10] M. Fischetti, F. Glover, A. Lodi. *The Feasibility Pump.* Mathematical Programming, 104, pp 91–104, 2005.

[11] M. Fischetti, A. Lodi. *Local Branching.* Mathematical Programming, 98(1-3), pp 23–47, 2003.

[12] M. Fischetti, D. Salvagnin. *Feasibility pump 2.0.* Mathematical Programming Computation, 1, pp 201–222, 2009.

[13] F. Glover, M. Laguna. *General purpose heuristics for integer programming part I.* Journal of Heuristics, 3, 1997.

[14] F. Glover, M. Laguna. *General purpose heuristics for integer programming part II.* Journal of Heuristics, 3, 1997.

[15] F. Glover, M. Laguna. *Tabu Search.* Kluwer Academic Publisher, Boston, Dordrecht, London, 1997.

[16] F. Glover, A. Løkketangen, D.L. Woodruff. *Scatter search to generate diverse MIP solutions.* in: M. Laguna, J. Gonzàlez-Velarde (Eds.), OR Computing Tools for Modeling, Optimization and Simulation: Interfaces in Computer Science and Operations Research, Kluwer Academic Publishers, pp. 299317, 2000.

[17] F.S. Hillier, R.M. Saltzman., *A heuristic ceiling point algorithm for general integer linear programming.* Management Science, 38(2), pp 263-283, 1992.

[18] A. Løkketangen, F. Glover. , *Solving zero/one mixed integer programming problems using tabu search.* European Journal of Operations Research, 106, pp 624-658, 1998.

[19] S. Lucidi, F. Rinaldi. *Exact penalty functions for nonlinear integer programming problems.* Journal of Optimization Theory and Applications Vol.145(3), 479-488, 2010.

[20] O. L. Mangasarian. *Solutions of General Linear Complementarity Problems via Nondifferentiable Concave Minimization.* Acta Mathematica Vietnamica, 22(1), pp 199–205, 1997.