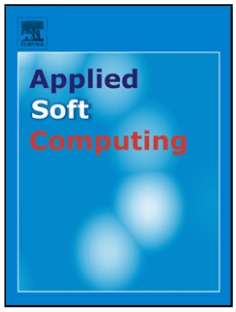


Title: A Local Search based Approach for Solving the Travelling Thief Problem

Author: Mohamed El Yafrani Belaïd Ahiod

PII: S1568-4946(16)30506-3
DOI: http://dx.doi.org/doi:10.1016/j.asoc.2016.09.047
Reference: ASOC 3846

To appear in: *Applied Soft Computing*

Received date: 4-4-2015
Revised date: 5-5-2016
Accepted date: 29-9-2016

Please cite this article as: Mohamed El Yafrani, Belaïd Ahiod, A Local Search based Approach for Solving the Travelling Thief Problem, <*![CDATA[Applied Soft Computing Journal]]*> (2016), http://dx.doi.org/10.1016/j.asoc.2016.09.047

**Algorithm 1** JNB (first-fit version)

$x \leftarrow$ *initial tour* ▷ we always use a Lin-Kernighan tour
$z \leftarrow$ *initial picking plan*
**repeat**
  Evaluate $G$, $t$, and $p$ for $(x, z)$
  Update $t^{map}$, $t^{acc}$, and $w^{acc}$
  **for** $i \leftarrow 2 \dots n-1$ **do**
    $\overline{x} \leftarrow swap(x, i)$
    $\overline{t^{map}} \leftarrow$ *tour mapper for* $(\overline{x}, z)$
    $\overline{t^{acc}} \leftarrow$ *recover time accumulator for* $(\overline{x}, z)$
    $\overline{w^{acc}} \leftarrow$ *recover weight accumulator for* $(\overline{x}, z)$
    **for** $k \leftarrow 1 \dots m$ **do**
      $\overline{z} \leftarrow bitflip(z, k)$
      **if** *no space left* **then** *skip iteration* **end if**
      *calculate* $\Delta_w$ *and* $\Delta_p$
      $\overline{p} \leftarrow p + \Delta_p$
      $i_{BF} \leftarrow t^{map}_{A_k}$ ▷ index of bit-flip
      **if** $i_{BF} = 0$ **then** $\overline{t} \leftarrow 0$ **else** $\overline{t} \leftarrow \overline{t^{acc}_{i_{BF}-1}}$ **end if**
      **for** $r \leftarrow i_{BF} \dots n$ **do**
        $w_c \leftarrow \overline{w^{acc}_r} + \Delta_w$
        $\overline{t} \leftarrow \overline{t} + \frac{d_{\overline{x}_r, \overline{x}_{r+1 mod n}}}{v_{max} - w_c * C}$
      **end for**
      $\overline{G} \leftarrow \overline{p} - R * \overline{t}$
      **if** $\overline{G} > G$ **then** *break loop* **end if** ▷ first fit
    **end for**
    **if** $\overline{G} > G$ **then** *break loop* **end if** ▷ first fit
  **end for**
  **if** $\overline{G} > G$ **then**
    $x \leftarrow \overline{x}$
    $z \leftarrow \overline{z}$
  **end if**
**until** $\overline{G} \leq G$