

## COLUMN GENERATION BASED HEURISTIC FOR THE GENERALIZED ASSIGNMENT PROBLEM

**Ruslan Sadykov**

INRIA Bordeaux — Sud-Ouest  
200 avenue de la Veille Tour, 33405 Talence, France  
ruslan.Sadykov@inria.fr

**François Vanderbeck**

Université Bordeaux, Institut de Mathématique de Bordeaux  
351 Cours de Libération, 33405 Talence, France  
fv@math.u-bordeaux1.fr

**Artur Pessoa**

**Eduardo Uchoa**

Universidade Federal Fluminense, Departamento de Engenharia de Produção  
Rua Passo da Pátria, 156, 24210-2140, Niterói, Brazil  
{pessoa, uchoa}@producao.uff.br

### RESUMO

Este artigo propõe uma heurística baseada em geração de colunas capaz de tratar instâncias grandes e difíceis do problema da atribuição generalizada. Inicialmente, apenas para conseguir obter a convergência da geração de colunas em tempos razoáveis, é utilizada uma sofisticada estabilização dual “tripla”, combinando funções de penalidade lineares por partes, o esquema de suavização dual proposto por Wentges e a suavização direcional por sub-gradiente. Para obter soluções boas inteiras viáveis, uma heurística de mergulho com diversidade é proposta. O método foi capaz de encontrar novas melhores soluções para todas as instâncias clássicas em aberto do problema de atribuição generalizada.

**PALAVRAS CHAVE.** Geração de colunas, Estabilização dual, Heurística de mergulho.

**Área Principal:** Programação Matemática

### ABSTRACT

This paper proposes a column generation based heuristic suitable for large and hard instances of the generalized assignment problem. First, just to be able to obtain the convergence of the column generation in reasonable times, it uses a sophisticated “triple” dual stabilization, combining piecewise linear penalty functions, Wentges’ dual smoothing and sub-gradient directional smoothing. In order to obtain integer good feasible solutions, a diversified diving heuristic is proposed. The overall method have found new best solutions for all open classic large-size instances of the generalized assignment problem.

**KEYWORDS.** Column generation, Dual stabilization, Diving heuristic.

**Main Area:** Mathematical Programming

## 1. Problem definition

In the generalized assignment problem, we need to assign  $n$  jobs to  $m$  machines (or agents). Each machine  $i$  has a given capacity,  $u_i$ . For each pair  $j = 1, \dots, n$  and  $i = 1, \dots, m$ , the assignment of job  $j$  to machine  $i$  consumes  $d_{ij}$  units of its capacity and contributes cost  $c_{ij}$  to the objective function. The problem consists in assigning each job to exactly one machine such that the capacities of machines are respected and the total cost is minimized. Using binary variables  $x_{ij}$  which determine whether job  $j$  is assigned to machine  $i$ , the problem can be modeled as the following Integer Program:

$$\min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (1)$$

$$\sum_{i=1}^m x_{ij} = 1, \quad \forall j = 1, \dots, n, \quad (2)$$

$$\sum_{j=1}^n d_{ij} x_{ij} \leq u_i, \quad \forall i = 1, \dots, m, \quad (3)$$

$$x_{ij} \in \{0, 1\}, \quad \forall i = 1, \dots, m, \forall j = 1, \dots, n \quad (4)$$

Here constraints (2) ensure that every job is assigned to some machine, and (3) guarantee that the capacity of each machine is respected.

Very often in the literature one works with the set covering reformulation of the problem. Here we use binary variables  $\lambda_g^i$  for each subset  $g \in G_i$  of jobs which can be simultaneously assigned to machine  $i$  without violating its capacity. Subsets  $g \in G_i$  are defined by values  $a^g$  such that  $a_{ij}^g = 1$  if job  $j$  is in set  $g$ , and  $a_{ij}^g = 0$  otherwise. The reformulation is then

$$\min \sum_{i=1}^m \sum_{g \in G_i} c_{ij} a_{ij}^g \lambda_g^i \quad (5)$$

$$\sum_{i=1}^m \sum_{g \in G_i} a_{ij}^g \lambda_g^i \geq 1, \quad \forall j = 1, \dots, n, \quad (6)$$

$$\sum_{g \in G_i} \lambda_g^i \leq 1, \quad \forall i = 1, \dots, m, \quad (7)$$

$$\lambda_g^i \in \{0, 1\}, \quad \forall i = 1, \dots, m, \forall g \in G_i. \quad (8)$$

Here constraints (6) ensure again that every job is put to some machine, and (7) guarantee that at most one subset of variables is assigned to each machine.

## 2. Literature review

There are two main exact approaches for the problem in the literature. The Branch-and-Cut approach is based on the compact formulation (1)–(4). The Branch-and-Price approach is based on the set-covering formulation (5)–(8). The first Branch-and-Price algorithm was proposed by [Savelsbergh, 1997]. The first Branch-and-Cut algorithm was suggested by [de Farias Jr. and Nemhauser, 2001], in which, in addition to standard knapsack valid inequalities, a specific family of inequalities was proposed.

[Pigatti et al., 2005] presented an improved Branch-and-Price algorithm. In order to improve a slow convergence of the column generation procedure embedded in the Branch-and-Price, the authors proposed a stabilization method based on 2-piece linear penalty functions. All but two OR-Library instances with up to 200 jobs were solved to optimality.

Later, [Avella et al., 2010] came up with an improved Branch-and-Cut approach, in which they proposed an algorithm for exact separation of knapsack valid inequalities, which are usually

separated only heuristically. They managed to solve all but 13 classic instances of the problem with up to 1,600 jobs.

The latest exact algorithm in the literature is due to [Posta et al., 2012]. In this work, the authors proposed a branch-and-bound algorithm based on a destructive Lagrangian dual bound. To obtain this bound, a bundle method is applied. To reduce the running time of the algorithm, the reduced cost variable fixing procedure is used. The proposed algorithm is able to solve all but 10 classic instances of the problem with up to 1,600 jobs. To our knowledge, these 10 instances remain open until now.

A heuristic algorithm which obtained the best published primal bounds for 8 of those open instances was presented by [Yagiura et al., 2006]. This algorithm is based on a path relinking approach with ejection chains.

### 3. Column generation

To solve the linear relaxation of the set covering reformulation (5)–(8), which is called master program, one can apply the column generation procedure. This procedure can be viewed as an application of the revised simplex algorithm with a specific procedure for pricing. One iteratively solves a version of the master program restricted to a subset of its variables, collects the dual solution  $\pi$ , and check the optimality of the current solution by pricing non-basic columns. Pricing is done by optimizing the reduced-cost value of solutions of the subproblem mixed integer polyhedra. In our case, the pricing problem is decomposed into binary knapsack subproblems, one for each machine  $i$ :

$$\min \sum_{j=1}^n (c_{ij} - \pi_j) \cdot x_j \quad (9)$$

$$\sum_{j=1}^n d_{ij} x_j \leq u_i, \quad (10)$$

$$x_j \in \{0, 1\}, \quad \forall j = 1, \dots, n. \quad (11)$$

This pricing problem is precisely that resulting from the Lagrangian relaxation of the complicating constraints (2). Hence, its solution allows to define a valid Lagrangian dual bound. Thus, the procedure yields a sequence of price vectors (dual solutions to the master program)  $\{\pi^t\}_t$ , where  $t$  denotes the iteration counter, converging towards  $\pi^* \in \Pi^*$ , where  $\Pi^*$  denotes the set of optimal dual solution to the master LP; and associated Lagrangian bound values,  $\{L(\pi^t)\}_t$ , converging towards the Lagrangian dual value  $L^*$ .

### 4. Stabilization

However, the textbook application of column generation to our problem suffers from very slow convergence. This is caused by the several drawbacks of the column generation procedure [Vanderbeck, 2005]:

- *Dual oscillations*: the dual values may oscillate widely in the first iterations. The “extreme columns” obtained from such extreme duals have no chance of appearing in an optimal solution.
- *Tailing-off effect*: in latter iterations, the added columns are likely to “cut” a marginal volume of the dual space, resulting in slow convergence.
- *Primal degeneracy and alternative optimal dual solutions*: the dual system has typically less active constraints than variables, thus admitting many alternative solutions. As a consequence, the column generation may pass by a number of such alternative dual solutions without making progress in the objective function.

Several stabilization techniques have been proposed to accelerate the convergence of sequence  $\{\pi^t\}_t$ , including Bundle methods [Oliveira and Sagastizábal] and interior-point methods [Gondzio et al., 2013]. In this work, we opted for only using families of techniques that allow one to continue to use the simplex algorithm for solving the restricted master problem.

Dual stabilization of [du Merle et al., 1999] is based on introduction of a piecewise linear function which penalizes deviation from the stability center, typically defined as the incumbent dual solution  $\hat{\pi}$ . The penalty function and the stability center are changing during the column generation procedure until the convergence to an optimal dual solution where the penalty function becomes null.

In [Pigatti et al., 2005], this stabilization technique was applied to the generalized assignment problem. The authors used the symmetric penalty function with 2 pieces. However, as it was showed by [Avella et al., 2010], this stabilization is still not sufficient for handling classic instances with more than 400 jobs.

Another family of stabilization techniques is based on the smoothing of dual values [Wentges, 1997]. Here, the separation point  $\tilde{\pi}^t$  (dual solution passed to the pricing problem) is calculated as  $\tilde{\pi}^t = \alpha\hat{\pi} + (1-\alpha)\pi^t$ . Parameter  $\alpha \in [0, 1)$  determines the position of the separation point on the line between stability center  $\hat{\pi}$  and the current dual solution  $\pi^t$  of the restricted master.

In [Pessoa et al., 2014], another kind of smoothing is proposed: the directional smoothing. In it, the separation point is moved towards a sub-gradient of the Lagrangian function at incumbent dual solution  $\hat{\pi}$ . Here parameter  $\beta \in [0, 1)$  determines how much the separation point is moved.

Following [Pessoa et al., 2014], we combine all the three mentioned stabilization techniques (penalty function with 3 pieces, Wentges smoothing, and directional smoothing) in order to improve convergence and reduce the running time of the column generation algorithm. Self-adjusting schemes are used for parameters  $\alpha$  and  $\beta$ , that control the degree of smoothing and directional smoothing applied to dual solutions, respectively. The only parameter  $\kappa$  of this combined stabilization technique is needed to determine the shape of the penalty function.

In Figure 1, we present comparison regarding running times of different stabilization techniques on 18 classic instances of the problem taken from [Yagiura et al., 2006]. Name of instances is in the format *class-m-n*. We used the following instances: D-20-100, E-20-100, D-10-100, E-10-100, D-20-200, E-20-200, D-40-400, E-40-400, D-5-100, E-5-100, D-10-200, E-10-200, D-20-400, E-20-400, D-5-200, E-5-200, D-10-400, E-10-400.

A point  $(x, y)$  on the performance profile associated to a stabilization scheme indicates that, for a fraction  $y$  of instances, the column generation algorithm with this scheme was not more than  $x$  times slower than the fastest algorithm for those instances.

One can see from the performance profiles that the combination of three stabilization techniques clearly outperforms all these techniques applied separately. Moreover, the “triple” stabilisation technique allowed us to solve the master problem for classic instances with 900 and 1,600 jobs for the first time.

## 5. Diversified diving heuristic

To obtain a feasible primal solution of the problem, a rounding heuristic procedure can be applied to the master LP solution. A standard rounding strategy consists of 2 steps: (i) an initial partial solution is obtained by rounding downwards the master LP solution; (ii) then, columns whose LP values are closest to the next integer are then considered for round up while feasible. However, reaching feasibility remains a difficult issue.

Diving heuristics are generic ways of “repairing” infeasibility. The residual master problem that remains after a rounding operation must be “cleanup” before re-optimization, deleting all columns that could not be part of an integer solution to the residual problem (and hence would lead to infeasibility if selected). Such preprocessing is a key feature in diving heuristics. It helps to avoid the primal heuristic dead-ending with an infeasible solution. Note that the re-optimization of

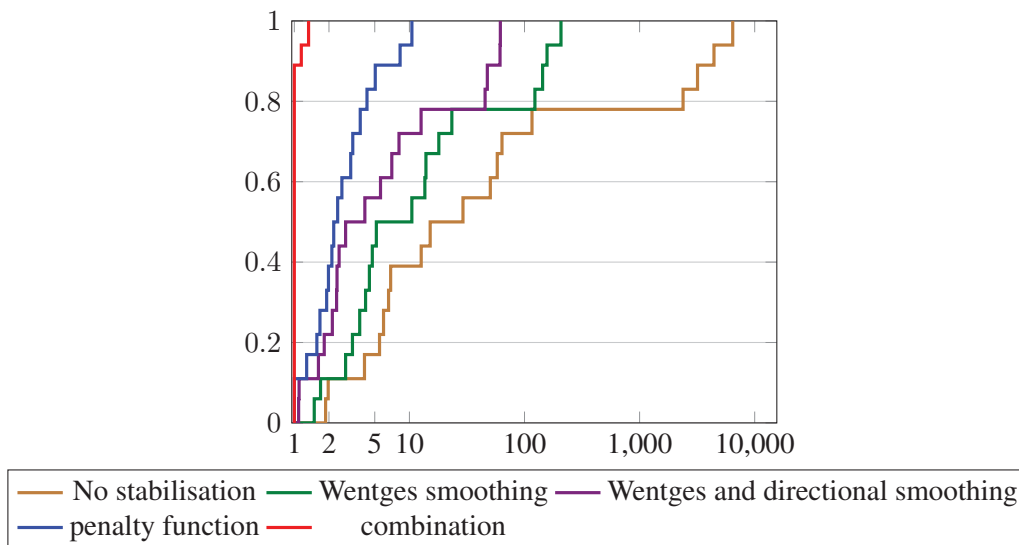


Figura 1: Performance profile for different stabilization techniques

the residual master might not necessarily be trivial and it can lead to generating new columns. This mechanism yields the “missing” complementary columns to build feasible solutions.

To add a diversification component to the diving heuristic, we rely on using the *Limited Discrepancy Search (LDS)* paradigm of [Harvey and Ginsberg, 2005], as proposed by [Joncour et al., 2010]. This search in the neighborhood of the reference (first dive) solution do have a great impact in improving the performance of the pure diving method.

In the search of better quality solution, the algorithm considers deviation from the reference solution up to depth  $maxDepth$ , in up  $maxDiscrepancy$  ways. Specifically, we avoid selecting columns present in a tabu list (of size  $\leq maxDiscrepancy$ ) that consists of columns selected in previous branches from which we wish to diversify the search. After the first (pure) diving, we backtrack until the current depth is equal to  $maxDepth$ . Then, we set the current branch as the only tabu in the tabu list to avoid selecting the corresponding column in the next dive, and we restart the diving from its ancestor node. The discrepancy of such diving will be exactly one. In a generic execution of the procedure, we backtrack while the current depth is greater than  $maxDepth$ , or the current discrepancy level =  $maxDiscrepancy$ . If such backtracking is not possible, the algorithm stops. Otherwise, after backtracking, a new branch is created that is defined by a tabu list made of columns that were tabu at the ancestor node or were selected in previous child nodes of the ancestor node. The resulting exploration tree is illustrated in Figure 2 for  $maxDepth = 3$  and  $maxDiscrepancy = 2$ .

## 6. Results

Using the column generation method with the “triple” stabilization technique coupled with the diversified diving heuristic, we improved the primal bounds published in the literature for all open instances from [Yagiura et al., 2006] of the problem. In our implementation, the stabilization parameter  $\kappa$  was set to 200. The parameters for diving heuristic with limited backtracking were set as following:  $maxDepth = 2$  and  $maxDiscrepancy = 3$ . The knapsack subproblems were solved by the MINKNAP algorithm proposed in [Pisinger, 1997].

The columns in Table 1 give the instance name (in the format *class-m-n*), the best known dual and primal bounds (taken from [Posta et al., 2012]), the new primal bounds found by our heuristic, and the time taken by it. Note that we have found an optimal solution for instance D-15-900 for the first time. That solution is shown in Table 2.

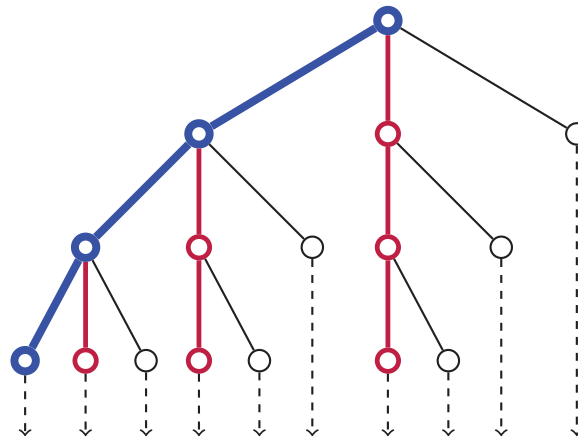


Figura 2: Diving heuristic with limited backtracking

| Instance  | Dual Bound | Previous BKS | New BKS      | Time   |
|-----------|------------|--------------|--------------|--------|
| D-20-200  | 12235      | 12244        | 12238        | 18s    |
| D-20-400  | 24563      | 24585        | 24568        | 1m03s  |
| D-40-400  | 24350      | 24417        | 24356        | 4m01s  |
| D-15-900  | 55404      | 55414        | <b>54404</b> | 8m52s  |
| D-30-900  | 54834      | 54868        | 54840        | 4m51s  |
| D-60-900  | 54551      | 54606        | 54554        | 14m09s |
| D-20-1600 | 97824      | 97837        | 97825        | 17m06s |
| D-40-1600 | 97105      | 97113        | 97106        | 15m19s |
| D-80-1600 | 97034      | 97052        | 97037        | 3h01m  |
| C-80-1600 | 16284      | 16289        | 16285        | 33m33s |

Tabela 1: Improved best known solutions obtained by the diversified diving heuristic

## Referências

- Avella, P., Boccia, M., and Vasilyev, I. (2010). A computational study of exact knapsack separation for the generalized assignment problem. *Computational Optimization and Applications*, 45(3):543–555.
- de Farias Jr., I. and Nemhauser, G. (2001). A family of inequalities for the generalized assignment polytope. *Operations Research Letters*, 29(2):49 – 55.
- Gondzio, J., González-Brevis, P., and Munari, P. (2013). New developments in the primal-dual column generation technique. *European Journal of Operational Research*, 224(1):41–51.
- William D. Harvey and Matthew L. Ginsberg. Limited discrepancy search. In *Proc. IJCAI-95, Montreal, Quebec*, pages 607–613. Morgan Kaufmann, 1995.
- Cédric Joncour, Sophie Michel, Ruslan Sadykov, Dmitry Sverdlov, and François Vanderbeck. Column generation based primal heuristics. *Electronic Notes in Discrete Mathematics*, 36:695 – 702, 2010.
- du Merle, O., Villeneuve, D., Desrosiers, J., and Hansen, P. (1999). Stabilized column generation. *Discrete Mathematics*, 194(1-3):229–237.



| Machine | Jobs                                                                                                                                                                                                                                                                                                                           |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1       | 2, 6, 8, 9, 15, 21, 24, 25, 29, 44, 60, 66, 70, 85, 87, 94, 118, 185, 202, 209, 210, 217, 223, 226, 258, 282, 293, 302, 331, 341, 366, 375, 395, 399, 405, 406, 444, 446, 453, 467, 476, 509, 513, 598, 643, 644, 660, 690, 698, 713, 714, 745, 746, 773, 793, 794, 808, 809, 818, 822, 828, 836, 860, 863, 870, 890           |
| 2       | 5, 38, 42, 58, 62, 65, 67, 112, 121, 131, 149, 151, 158, 184, 200, 201, 205, 206, 211, 219, 230, 287, 303, 344, 359, 390, 429, 441, 450, 466, 479, 498, 541, 551, 553, 557, 558, 588, 591, 606, 617, 620, 646, 650, 658, 693, 695, 701, 712, 723, 733, 741, 742, 747, 756, 784, 791, 830, 850, 851, 852                        |
| 3       | 1, 12, 45, 55, 63, 110, 116, 129, 169, 178, 179, 198, 199, 249, 254, 261, 278, 279, 309, 310, 326, 327, 351, 354, 362, 364, 381, 387, 389, 397, 408, 430, 443, 482, 492, 535, 552, 585, 587, 595, 611, 625, 637, 711, 728, 731, 738, 759, 787, 800, 824, 832, 835, 840, 844, 893, 895                                          |
| 4       | 33, 41, 49, 53, 81, 98, 108, 119, 146, 154, 155, 177, 180, 203, 222, 227, 231, 250, 270, 283, 286, 328, 338, 347, 363, 368, 373, 382, 383, 384, 393, 412, 452, 472, 503, 530, 561, 571, 581, 602, 628, 630, 633, 634, 663, 665, 717, 721, 732, 740, 748, 763, 775, 781, 788, 812, 857, 861, 877, 881, 888, 898                 |
| 5       | 40, 50, 61, 88, 107, 130, 138, 161, 168, 176, 207, 221, 224, 241, 275, 277, 290, 297, 298, 304, 321, 335, 342, 361, 378, 415, 418, 426, 431, 470, 518, 527, 548, 564, 603, 615, 626, 651, 653, 670, 675, 689, 716, 719, 778, 819, 820, 834, 845, 847, 862, 882                                                                 |
| 6       | 27, 34, 36, 39, 52, 71, 79, 122, 128, 132, 134, 143, 183, 193, 208, 232, 233, 236, 301, 308, 311, 316, 323, 337, 343, 358, 367, 391, 403, 404, 421, 447, 448, 460, 465, 477, 500, 532, 572, 580, 601, 607, 609, 618, 638, 639, 648, 649, 657, 672, 704, 749, 802, 846, 854, 871, 883, 889                                      |
| 7       | 23, 32, 64, 68, 80, 95, 135, 162, 172, 196, 228, 239, 244, 247, 259, 268, 276, 281, 318, 325, 356, 392, 396, 401, 414, 461, 480, 506, 522, 525, 539, 542, 559, 599, 613, 624, 629, 654, 656, 669, 687, 696, 699, 735, 744, 762, 766, 796, 811, 817, 827, 839, 884, 900                                                         |
| 8       | 17, 20, 37, 46, 48, 51, 57, 103, 106, 152, 187, 188, 237, 245, 253, 264, 285, 332, 352, 353, 369, 379, 424, 433, 462, 468, 473, 475, 485, 502, 511, 514, 529, 556, 567, 573, 574, 577, 582, 590, 610, 635, 641, 645, 671, 678, 685, 707, 729, 730, 753, 755, 758, 774, 777, 780, 810, 843, 849, 856, 864, 876, 880, 886, 896   |
| 9       | 3, 14, 35, 54, 73, 74, 104, 109, 117, 133, 137, 142, 145, 150, 160, 170, 171, 173, 204, 213, 215, 220, 234, 238, 272, 284, 313, 333, 340, 355, 372, 428, 435, 451, 454, 478, 483, 487, 494, 524, 534, 537, 547, 562, 570, 586, 605, 622, 659, 667, 679, 686, 703, 727, 736, 768, 792, 803, 866, 867, 875, 891                  |
| 10      | 7, 59, 77, 82, 83, 102, 123, 139, 140, 147, 164, 174, 197, 216, 248, 263, 265, 271, 291, 295, 299, 312, 336, 345, 365, 374, 398, 413, 419, 425, 439, 504, 507, 520, 521, 526, 533, 560, 565, 575, 604, 619, 655, 662, 676, 677, 680, 691, 697, 705, 725, 737, 754, 757, 779, 816, 841, 887                                     |
| 11      | 10, 26, 31, 43, 76, 90, 92, 105, 120, 127, 156, 167, 181, 191, 255, 306, 307, 315, 334, 349, 357, 371, 409, 420, 456, 459, 463, 508, 512, 517, 519, 528, 554, 578, 623, 631, 640, 642, 664, 681, 694, 722, 734, 743, 752, 770, 782, 786, 789, 790, 798, 799, 804, 807, 815, 821, 825, 842, 848, 853, 865, 869, 879             |
| 12      | 28, 30, 78, 86, 91, 141, 153, 157, 165, 166, 175, 189, 195, 242, 246, 273, 280, 288, 296, 300, 314, 317, 320, 324, 329, 339, 348, 350, 370, 394, 402, 423, 436, 437, 438, 445, 488, 495, 499, 516, 544, 549, 555, 566, 569, 596, 597, 627, 652, 682, 683, 688, 706, 708, 710, 760, 765, 769, 785, 795, 813, 833, 858, 874, 897 |
| 13      | 4, 11, 13, 18, 19, 22, 84, 96, 97, 101, 114, 124, 136, 148, 194, 229, 235, 252, 322, 346, 416, 417, 440, 486, 489, 491, 501, 505, 510, 531, 536, 540, 550, 579, 583, 584, 593, 600, 612, 616, 632, 661, 673, 692, 702, 720, 724, 750, 764, 767, 783, 797, 805, 814, 831, 838, 855, 873                                         |
| 14      | 16, 47, 69, 72, 89, 93, 100, 115, 125, 126, 144, 163, 186, 190, 192, 212, 214, 218, 225, 243, 256, 257, 260, 262, 267, 289, 292, 305, 319, 330, 360, 376, 377, 380, 385, 386, 410, 427, 469, 471, 481, 490, 493, 523, 543, 545, 546, 563, 608, 621, 647, 684, 715, 751, 761, 771, 859, 885, 892, 899                           |
| 15      | 56, 75, 99, 111, 113, 159, 182, 240, 251, 266, 269, 274, 294, 388, 400, 407, 411, 422, 432, 434, 442, 449, 455, 457, 458, 464, 474, 484, 496, 497, 515, 538, 568, 576, 589, 592, 594, 614, 636, 666, 668, 674, 700, 709, 718, 726, 739, 772, 776, 801, 806, 823, 826, 829, 837, 868, 872, 878, 894                             |

Tabela 2: Optimal solution for the instance D-15-900

Oliveira, W., and Sagastizábal, C. (2014). Bundle methods in the XXIst century: A bird's-eye view. *Pesquisa Operacional*, 34:647–670.

Pessoa, A., Sadykov, R., Uchoa, E., and Vanderbeck, F. (2014). Automation and combination of linear-programming based stabilization techniques in column generation. Technical Report hal-01077984, HAL Inria, <https://hal.inria.fr/hal-01077984>.

Pigatti, A., Poggi de Aragão, M., and Uchoa, E. (2005). Stabilized branch-and-cut-and-price for the generalized assignment problem. *Electronic Notes in Discrete Mathematics*, 19:389 – 395.

- Pisinger, D. (1997). A minimal algorithm for the 0-1 knapsack problem. *Operations Research*, 45(5):758–767.
- Posta, M., Ferland, J. A., and Michelon, P. (2012). An exact method with variable fixing for solving the generalized assignment problem. *Computational Optimization and Applications*, 52:629–644.
- Savelsbergh, M. (1997). A branch-and-price algorithm for the generalized assignment problem. *Operations Research*, 45(6):831–841.
- Vanderbeck, F. (2005). Implementing mixed integer column generation. In *Column Generation* (pp. 331-358). Springer.
- Wentges, P. (1997). Weighted dantzig–wolfe decomposition for linear mixed-integer programming. *International Transactions in Operational Research*, 4(2):151–162.
- Yagiura, M., Ibaraki, T., and Glover, F. (2006). A path relinking approach with ejection chains for the generalized assignment problem. *European Journal of Operational Research*, 169(2):548 – 569.

