Machine Learning-based Bundle Method

Demelas F.[1,2], Frangioni A.[2], Lacroix M.[1], Le Roux J.[1], Traversi E.[3], Wolfler Calvo R[1].

[1] LIPN, Université Sorbonne Paris Nord - CNRS
{demelas, lacroix,leroux}@lipn.univ-paris13.fr
[2] Dipartimento di Informatica, Università di Pisa
frangioni@unipi.it
[3] ESSEC, Business School
emiliano.traversi@essec.edu

Keywords : *Bundle Method, Lagrangian Relaxation, Recurrent Neural Network, Unrolling*

Given a Mixed Integer Linear Program (MILP):

$$(P)\ \max_{\boldsymbol{x}\in X} \boldsymbol{c}^\top \boldsymbol{x} \text{ with } X = \{\boldsymbol{x} \in \mathbb{R}^{m_1} \times \mathbb{N}^{m_2} \ :\ \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b},\ \boldsymbol{C}\boldsymbol{x} \leq \boldsymbol{d}\},$$

Lagrangian relaxation (LR) [5] is a well-known optimization method that reduces the complexity of the problem by relaxing some hard inequalities $\boldsymbol{C}\boldsymbol{x} \leq \boldsymbol{d}$. Relaxed constraints are incorporated into the objective function as a soft penalty whose coefficients are referred to as *Lagrangian multipliers.* For each given vector of Lagrangian multiplier $\boldsymbol{\pi} \geq 0$, this gives the *Lagrangian problem* whose optimum provides an upper bound for the original problem

$$\phi(\boldsymbol{\pi}) = \max_{\boldsymbol{x}\in\mathbb{R}^{m_1}\times\mathbb{N}^{m_2}\,:\,\boldsymbol{A}\boldsymbol{x}\leq\boldsymbol{b}} \boldsymbol{c}^\top\boldsymbol{x} + (\boldsymbol{d} - \boldsymbol{C}\boldsymbol{x})^\top\boldsymbol{\pi}.$$

Hence, we can consider a convex piece-wise linear function $\phi : \mathbb{R}^n_+ \to \mathbb{R}$ associating to each vector of Lagrangian multiplier the corresponding bound. The *Lagrangian dual* problem (LD) consists of determining the Lagrangian multipliers providing the tightest bound:

$$\min_{\boldsymbol{\pi}\in\mathbb{R}^n_+} \phi(\boldsymbol{\pi}).$$

This type of problem can be solved using Bundle methods (BMs) [8]. BMs iteratively tries to update the stabilization point, that is, the best point found so far, using a *search direction* obtained as the one minimizing an approximation of the function ϕ provided by a *surrogate model* iteratively refined using the obtained information. In this case, a common choice of surrogate model is the cutting planes model [11].

Machine Learning (ML) has recently been widely used in the domain of combinatorial optimization [3]. In this work, we propose to enhance BM with ML. More precisely, exploiting unrolling techniques [9], we substitute the resolution of the (penalized) convex problem, providing the search direction, with the prediction of an ML model based on a Long-Short Term Memory Network [6] and an Attention Network[13]. This corresponds to using an ML model as a (regularized) *surrogate model* and makes the execution of the algorithm differentiable. This is similar to what is done in [2] for gradient descent. We train our network to minimize the value of ϕ in the last point $\boldsymbol{\pi}_T$ found after a fixed amount T of iterations. To our knowledge, ML has not yet been directly applied within the BMs, but it has been used to predict high-quality LM [4, 12].

We test our approach on two different LRs: one for the Multi-Commodity Fixed-Charge Network Design Problem (MC) [1] and the other for the Generalized Assignment Problem (GA) [10]. For both problems, we consider further datasets, some of which are composed of small (-SML) or large (-BIG) instances. Numerical results, summarized in Table 1, show that our model can provide a high-quality solution, often better than the best baselines, obtained by considering several sub-gradients and BMs for which the starting regularization parameter/step-size is chosen with grid search. Furthermore, the model shows good generalization properties for more iterations than the ones for which the model was trained.

These experiments show the potential of ML to enhance BM. Other experiments should be conducted, especially in Nonlinear Programming problems like the ones arising from the ML community [7].

TAB. 1: Comparison of our approach with the best baseline, for which the starting regularization parameter/step-size is chosen with grid search for several sub-gradient and bundle methods. Our approach does not need hyperparameter tuning at inference, but it is trained on 10 iterations.

	Methods	10 iter.		25 iter.		50 iter.		100 iter.	
		GAP	time	GAP	time	GAP	time	GAP	time
MC-SML	Best Baseline	11.96	0.048	4.16	0.128	1.30	0.293	0.31	0.816
	Bundle Network	**9.20**	0.105	**1.75**	0.185	**0.53**	0.321	**0.17**	0.592
MC-BIG	Best Baseline	**17.98**	0.112	6.34	0.289	2.66	0.627	0.93	1.477
	Bundle Network	20.10	0.139	**5.11**	0.271	**2.01**	0.495	**0.70**	0.953
GA-SML	Best Baseline	0.1893	0.071	**0.0156**	0.211	**0.0014**	0.601	**0.0006**	1.596
	Bundle Network	**0.1484**	0.104	0.0228	0.177	0.0047	0.304	0.0009	0.551
GA-BIG	Best Baseline	0.1129	0.433	0.0193	1.459	0.0052	4.264	0.0020	14.459
	Bundle Network	**0.1018**	0.228	**0.0190**	0.498	0.0052	0.943	**0.0014**	1.837

References

[1] Mohammad Rahim Akhavan Kazemzadeh, Tolga Bektaş, Teodor Gabriel Crainic, Antonio Frangioni, Bernard Gendron, and Enrico Gorgone. Node-based lagrangian relaxations for multicommodity capacitated fixed-charge network design. *Discrete Applied Mathematics*, 308:255–275, 2022. Combinatorial Optimization ISCO 2018.

[2] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W. Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando de Freitas. Learning to learn by gradient descent by gradient descent, 2016.

[3] Yoshua Bengio, Andrea Lodi, and Antoine Prouvost. Machine learning for combinatorial optimization: A methodological tour d'horizon. *European Journal of Operational Research*, 290(2):405–421, 2021.

[4] Francesco Demelas, Joseph Le Roux, Mathieu Lacroix, and Axel Parmentier. Predicting lagrangian multipliers for mixed integer linear programs. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pages 10368–10384. PMLR, 21–27 Jul 2024.

[5] Antonio Frangioni. About lagrangian methods in integer optimization. *Ann Oper Res*, 139(1):163–193, 2005.

[6] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997.

[7] Quoc Le, Alex Smola, and Svn Vishwanathan. Bundle methods for machine learning. *Advances in neural information processing systems*, 20, 2007.

[8] Claude Lemarechal, Jean-Jacques Strodiot, and André Bihain. On a bundle algorithm for nonsmooth optimization. In *Nonlinear programming 4*, pages 245–282. Elsevier, 1981.

[9] Vishal Monga, Yuelong Li, and Yonina C Eldar. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44, 2021.

[10] Temel Öncan. A survey of the generalized assignment problem and its applications. *INFOR: Information Systems and Operational Research*, 45(3):123–141, 2007.

[11] Alexander Schrijver et al. On cutting planes. *Combinatorics*, 79:291–296, 1980.

[12] Nagisa Sugishita, Andreas Grothey, and Ken McKinnon. Use of machine learning models to warmstart column generation for unit commitment. *INFORMS Journal on Computing*, 2024.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.