

Approximation algorithms for integer programming with resource augmentation

Hauke Brinkop¹, Hua Chen², Lin Chen³, Klaus Jansen¹, and Guochuan Zhang³

¹Kiel University, Germany

²Zhejiang University of Technology, Hangzhou, China

³Zhejiang University, Hangzhou, China

Abstract

Integer programs (IPs) find widespread applications in many combinatorial optimization problems. Generally, an integer program can be formulated as follows:

$$\min\{\mathbf{w}\mathbf{x} : H\mathbf{x} = \mathbf{b}, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}, \mathbf{x} \in \mathbb{Z}^n\}.$$

Solving a general IP is NP-hard. The classic algorithm [Papadimitriou, J.ACM '81] for IPs has a running time $n^{O(m)}(m \cdot \max\{\Delta, \|\mathbf{b}\|_\infty\})^{O(m^2)}$ [8], where m is the number of constraints, n is the number of variables, and Δ and $\|\mathbf{b}\|_\infty$ are, respectively, the largest absolute values among the entries in the constraint matrix and the right-hand side vector of the constraint. The running time is exponential in m , and becomes pseudo-polynomial if m is a constant. In recent years, there has been extensive research on FPT (fixed parameter tractable) algorithms for the so-called n -fold IPs, which may possess a large number of constraints, but the constraint matrix satisfies a specific block structure. It is remarkable that these FPT algorithms take as parameters Δ and the number of rows and columns of some small submatrices. If Δ is not treated as a parameter, then the running time becomes pseudo-polynomial even if all the other parameters are taken as constants. However, a polynomial time algorithm may be possible if we are allowed to solve an IP approximately in the sense that the constraints can be violated slightly.

One most well-known example is Knapsack, where H contains one row, i.e., $m = 1$, $\mathbf{l} = \mathbf{0}$, $\mathbf{u} = \mathbf{1}$, and all entries of \mathbf{w} and H are nonnegative. Knapsack admits an FPTAS whose running time is polynomial in the input length $|I|$ and $1/\varepsilon$ instead of Δ (see, e.g., [5]). Alternatively, it admits an algorithm of the same running time that returns a solution whose objective value is at least OPT (where OPT denotes the optimal objective value satisfying the constraint), while violating the constraint by a factor of $1 + \varepsilon$.

Another example is due to Dadush [3], who presented an algorithm of running time $2^{O(n)}(1/\varepsilon^2)^n$ which either asserts that the convex body K described by the constraints does not contain any integer points, or finds an integer point in the body stemming from K scaled by $1 + \varepsilon$ from its center of gravity.

Motivated by the above two examples, this paper aims to explore approximation algorithms for IPs. We are interested in an algorithm that returns a near-feasible solution that may violate the constraints by a factor of $1 + \varepsilon$, but runs in polynomial time in the input. Such relaxation of constraints is also referred to as *resource augmentation*, which has received much attention in various combinatorial optimization problems, including Knapsack [1, 6], Subset Sum [2, 7], Square Packing [4], etc.

Main Results. Our first result is an approximation scheme for IPs with few constraints.

Theorem 1. *Given is an integer program $\min\{\mathbf{w}\mathbf{x} : H\mathbf{x} = \mathbf{b}, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}, \mathbf{x} \in \mathbb{Z}^n\}$ with optimal objective value OPT, where $H \in \mathbb{Q}^{m \times n}$. Then for arbitrary small $\varepsilon > 0$, there exists an algorithm of running time $f(m, \varepsilon) \cdot \text{poly}(|I|)$ which returns a near-feasible solution $\tilde{\mathbf{x}}$ such that $\tilde{\mathbf{x}} \in \{\mathbf{x} : \|H\mathbf{x} - \mathbf{b}\|_\infty \leq \varepsilon\Delta, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}, \mathbf{x} \in \mathbb{Z}^n\}$, and $\mathbf{w}\tilde{\mathbf{x}} \leq \text{OPT}$, where $\Delta = \|H\|_\infty$.*

We remark that OPT always refers to the optimal objective value **without** violating the constraints.

In particular, the running time in Theorem 1 is $2^{(\frac{m}{\varepsilon})^{O(m)}} \cdot \text{poly}(|I|)$. Theorem 1 extends Papadimitriou's algorithm in the direction of approximation. We then extend our result to n -fold IPs and obtain the following.

Theorem 2. Given is an integer program $\min\{\mathbf{w}\mathbf{x} : \sum_{i=1}^n D^i \mathbf{x}^i = \mathbf{b}^0, A^i \mathbf{x}^i = \mathbf{b}^i, 1 \leq i \leq n, \mathbf{0} \leq \mathbf{x} \leq \mathbf{u}, \mathbf{x} \in \mathbb{Z}^m\}$ with optimal objective value OPT, where $A^i \in \mathbb{Q}_{\geq 0}^{s_A \times t_A}$, $D^i \in \mathbb{Q}_{\geq 0}^{s_D \times t_D}$, and $t_A = t_D = t$. Then for arbitrary small $\varepsilon > 0$, there exists an algorithm of running time $f(s_A, s_D, t, \varepsilon) \cdot \text{poly}(|I|)$ which returns a near-feasible solution $\tilde{\mathbf{x}}$ such that $\tilde{\mathbf{x}} \in \{\mathbf{x} : (1 - \varepsilon)\mathbf{b}^0 \leq \sum_{i=1}^n D^i \mathbf{x}^i \leq (1 + \varepsilon)\mathbf{b}^0, (1 - \varepsilon)\mathbf{b}^i \leq A^i \mathbf{x}^i \leq (1 + \varepsilon)\mathbf{b}^i, 1 \leq i \leq n, \mathbf{0} \leq \mathbf{x} \leq \mathbf{u}, \mathbf{x} \in \mathbb{Z}^m\}$, and $\mathbf{w}\tilde{\mathbf{x}} \leq \text{OPT}$.

In particular, the running time is $2^{2^{(s_D/\varepsilon)^{O(s_D)} \cdot (s_A t/\varepsilon)^{O(t)}}} \cdot \text{poly}(|I|)$. Theorem 2 is a bit weak in the sense that it requires all the entries of the submatrices A^i, D^i as well as the variables to be nonnegative, and it gives an additive error of $\varepsilon \mathbf{b}^i$ instead of $\varepsilon \Delta$. In many combinatorial optimization problems where n -fold IPs are applicable, all the input parameters are indeed nonnegative and $(1 + \varepsilon)\mathbf{b}^i$ yields a standard multiplicative $(1 + \varepsilon)$ -factor. Nevertheless, we show that it is possible to get rid of such a weakness if the local constraint $A^i \mathbf{x}^i = \mathbf{b}^i$ has few solutions, as implied by the following Theorem 3.

Theorem 3. Given is an integer program $\min\{\mathbf{w}\mathbf{x} : \sum_{i=1}^n D^i \mathbf{x}^i = \mathbf{b}^0, \mathbf{x}^i \in \mathcal{P}^i, 1 \leq i \leq n, \mathbf{x} \in \mathbb{Z}^m\}$ with optimal objective value OPT, where $D^i \in \mathbb{Q}^{s \times t}$ and \mathcal{P}^i is an arbitrary set of integer vectors. Let $\kappa = \max_{\mathbf{u} \in \cup_{i=1}^n \mathcal{P}^i} \|\mathbf{u}\|_\infty$, i.e., κ is the largest ℓ_∞ -norm among all integer vectors in $\cup_{i=1}^n \mathcal{P}^i$. Then for arbitrary small $\varepsilon > 0$, there exists an algorithm of running time $f(s, t, \kappa, \varepsilon) \cdot \text{poly}(|I|)$ which returns a near-feasible solution $\tilde{\mathbf{x}}$ such that $\tilde{\mathbf{x}} \in \{\mathbf{x} : \|\sum_{i=1}^n D^i \mathbf{x}^i - \mathbf{b}^0\|_\infty \leq \varepsilon \Delta, \mathbf{x}^i \in \mathcal{P}^i, 1 \leq i \leq n, \mathbf{x} \in \mathbb{Z}^m\}$, and $\mathbf{w}\tilde{\mathbf{x}} \leq \text{OPT}$, where $\Delta = \max_{i \in [n]} \|D^i\|_\infty$.

In particular, the running time is $2^{(\frac{s\kappa^t}{\varepsilon})^{O(s\kappa^t)}} \cdot \text{poly}(|I|)$. Theorem 2 and Theorem 3 can be viewed as FPT approximation schemes for n -fold IPs with different parameters.

References

- [1] K. Bringmann and A. Cassis. Faster knapsack algorithms via bounded monotone min-plus-convolution. In *Proceedings of the 49th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 229 of *LIPICs*, pages 31:1–31:21, 2022.
- [2] L. Chen, J. Lian, Y. Mao, and G. Zhang. Approximating partition in near-linear time. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC)*, pages 307–318, 2024.
- [3] D. Dadush. A randomized sieving algorithm for approximate integer programming. *Algorithmica*, 70(2):208–244, 2014.
- [4] A. V. Fishkin, O. Gerber, K. Jansen, and R. Solis-Oba. On packing squares with resource augmentation: Maximizing the profit. In *Proceedings of the 2005 Australasian Symposium on Theory of Computing-Volume 41*, pages 61–67, 2005.
- [5] O. H. Ibarra and C. E. Kim. Fast approximation algorithms for the knapsack and sum of subset problems. *Journal of the ACM*, 22(4):463–468, 1975.
- [6] K. Iwama and G. Zhang. Online knapsack with resource augmentation. *Information Processing Letters*, 110(22):1016–1020, 2010.
- [7] M. Mucha, K. Węgrzycki, and M. Włodarczyk. A subquadratic approximation scheme for partition. In *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 70–88. SIAM, 2019.
- [8] C. H. Papadimitriou. On the complexity of integer programming. *Journal of the ACM*, 28(4):765–768, 1981.