ZONOTOPES

THOMAS WINNINGER

Abstract. Zonotopes are promising abstract domains for machine learning oriented tasks due to their efficiency, but they fail to capture complex transformations needed in new architectures, like the softmax. To overcome this issue, recent work proposed more precise zonotopes, like hybrid constrained zonotopes, or polynomial zonotopes. However, these precise zonotopes often require solving MILP problems at each step, which makes them unusable. This work - focused on the recent transformer architecture - aims to unite different zonotopes methods to reduce the computational overload, while maintaining sufficient precision.

! *work in progress* !

1. Classical Zonotope

A classical Zonotope [1] abstracts a set of $N \in \mathbb{N}$ variables and associates the k -th variable with an affine expression x_k using $I \in \mathbb{N}$ noise symbols defined by:

$$x_k = c_k + \sum_{i=1}^{I} \gamma_{ik}\varepsilon_i = c_k + G_k\mathcal{E} \tag{1}$$

where $c_k, \gamma_{ik} \in \mathbb{R}$ and $\varepsilon_i \in [-1, 1]$. The value x_k can deviate from its center coefficient c_k through a series of noise symbols ε_i scaled by the coefficients γ_{ik}. The set of noise symbols $\mathcal{E}$ is shared among different variables, thus encoding dependencies between N values abstracted by the zonotope.

2. Hybrid constrained zonotope

The hybrid constrained zonotope is defined as:

$$z = c + G\mathcal{E} + G'\mathcal{E}', \qquad \text{s.t. } \begin{cases} A\mathcal{E} + A'\mathcal{E}' = b \\ \mathcal{E} \in [-1, 1]^I \\ \mathcal{E}' \in \{-1, 1\}^{I'} \end{cases} \tag{2}$$

With N the number of variables, I the number of continuous noise terms, I' the number of binary noise terms, J the number of constraints, $z, c \in \mathbb{R}^n$, $G \in \mathbb{R}^n \times \mathbb{R}^I$, $G' \in \mathbb{R}^n \times \mathbb{R}^{I'}$, $A \in \mathbb{R}^J \times \mathbb{R}^I$, $A' \in \mathbb{R}^J \times \mathbb{R}^{I'}$, $b \in \mathbb{R}^J$.

2.1. Concretisation.

The lower bound is defined as:

$$l = \min_{\substack{A\mathcal{E}+A'\mathcal{E}'=b \\ \mathcal{E}\in[-1,1]^I \\ \mathcal{E}'\in\{-1,1\}^{I'}}} c + G\mathcal{E} + G'\mathcal{E}' \tag{3}$$

This is a minimisation problem that can be solved with a MILP, which would make compute complexity grow exponentially. However, it is possible to compute sound bounds with relative precision considering the dual problem.

Using Lagrange multipliers, the previous minimisation problem can be rewritten:

$$l = \min_{\substack{\mathcal{E}\in[-1,1]^I \\ \mathcal{E}'\in\{-1,1\}^{I'}}} \max_{\lambda\in\mathbb{R}^J} c + G\mathcal{E} + G'\mathcal{E}' - \sum_j^J \lambda_j(A_j\mathcal{E} + A'_j\mathcal{E}' - b_j) \tag{4.1}$$

$$= \min_{\substack{\mathcal{E}\in[-1,1]^I \\ \mathcal{E}'\in\{-1,1\}^{I'}}} \max_{\Lambda\in\mathbb{R}^N\times\mathbb{R}^J} c + \Lambda b - (G - \Lambda A)\mathcal{E} - (G' - \Lambda A')\mathcal{E}' \tag{4.2}$$

Since the objective is linear and the optimisation variables are compact ($[-1,1]^I \times \{-1,1\}^{I'}$), we can reverse the order of the min and the max and obtain the same bound:

$$l = \max_{\Lambda\in\mathbb{R}^N\times\mathbb{R}^J} \min_{\substack{\mathcal{E}\in[-1,1]^I \\ \mathcal{E}'\in\{-1,1\}^{I'}}} c + \Lambda b - (G - \Lambda A)\mathcal{E} - (G' - \Lambda A')\mathcal{E}' \tag{5.1}$$

$$= \max_{\Lambda\in\mathbb{R}^N\times\mathbb{R}^J} c + \Lambda b - \|G - \Lambda A\|_1 - \|G' - \Lambda A'\|_1 \tag{5.2}$$

$$= \max_{\Lambda\in\mathbb{R}^N\times\mathbb{R}^J} d(\Lambda) \tag{5.3}$$

We can observe that, for any Λ, $d(\Lambda) \leq l$ **(TODO: Verify)**. With $\Lambda = 0$, it becomes the concretisation of the classical zonotope. Thus, $d(\Lambda)$ is a sound bound that can be optimised.

2.2. **Operations [2].**

For $Z = \langle c_z, G_z, G'_z, A_z, A'_z, b_z\rangle \in \mathbb{R}^N, Y = \langle c_y, G_y, G'_y, A_y, A'_y, b_y\rangle \in \mathbb{R}^N, W = \langle c_w, G_w, G'_w, A_w, A'_w, b_w\rangle \in \mathbb{R}^M, R \in \mathbb{R}^{M\times N}$:

$$RZ = \langle Rc_z, RG_z, RG'_z, A_z, A'_z, b_z\rangle \tag{6}$$

Minkowski sums:

$$Z \oplus Y = \left\langle c_z + c_y, \begin{bmatrix} G_z & G_y \end{bmatrix}, \begin{bmatrix} G'_z & G'_y \end{bmatrix}, \begin{bmatrix} A_z & 0 \\ 0 & A_y \end{bmatrix}, \begin{bmatrix} A'_z & 0 \\ 0 & A'_y \end{bmatrix}, \begin{bmatrix} b_z \\ b_y \end{bmatrix} \right\rangle \tag{7}$$

Intersection:

$$Z \cap_R Y = \left\langle c_z, \begin{bmatrix} G_z & \mathbf{0} \end{bmatrix}, \begin{bmatrix} G_y & \mathbf{0} \end{bmatrix}, \begin{bmatrix} A_z & \mathbf{0} \\ \mathbf{0} & A_y \\ RG_z & -G_y \end{bmatrix}, \begin{bmatrix} A'_z & \mathbf{0} \\ \mathbf{0} & A'_y \\ RG'_z & -G'_y \end{bmatrix}, \begin{bmatrix} b_z \\ b_y \\ c_y - Rc_z \end{bmatrix} \right\rangle \tag{8}$$

Union, $Z \cup Y$:

$$c_u = \frac{1}{2}(c_z + c_y + G'_z\mathbf{1} + G'_y\mathbf{1}) \tag{9.1}$$

$$\hat{G}' = \frac{1}{2}(c_z - c_y + G'_y\mathbf{1} - G'_z\mathbf{1}) \tag{9.2}$$

$$\hat{A}'_z = -\frac{1}{2}(b_z + A'_z\mathbf{1}),\ \hat{b}_z = \frac{1}{2}(b_z - A'_z\mathbf{1}) \tag{9.3}$$

$$\hat{A}'_y = \frac{1}{2}(b_y + A'_y\mathbf{1}),\ \hat{b}_y = \frac{1}{2}(b_y - A'_y\mathbf{1}) \tag{9.4}$$

$$G_u = \begin{bmatrix} G_z & G_y & \mathbf{0} \end{bmatrix},\ G'_u = \begin{bmatrix} G'_z & G'_y & \hat{G}' \end{bmatrix} \tag{9.5}$$

$$A_u = \begin{bmatrix} A_z & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & A_y & \mathbf{0} \\ & A_3 & \mathbf{I} \end{bmatrix},\ A'_u = \begin{bmatrix} A'_z & \mathbf{0} & \hat{A}'_z \\ \mathbf{0} & A'_y & \hat{A}'_y \\ & A'_3 & \end{bmatrix},\ b_u = \begin{bmatrix} \hat{b}_z \\ \hat{b}_y \\ b_3 \end{bmatrix} \tag{9.6}$$

$$A_3 = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \\ \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix},\ A'_3 = \frac{1}{2}\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & -\mathbf{1} \\ \mathbf{0} & \mathbf{0} & -\mathbf{1} \\ \mathbf{I} & \mathbf{0} & \mathbf{1} \\ -\mathbf{I} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{I} & -\mathbf{1} \\ \mathbf{0} & -\mathbf{I} & -\mathbf{1} \end{bmatrix},\ b_3 = \begin{bmatrix} \frac{1}{2}\mathbf{1} \\ \frac{1}{2}\mathbf{1} \\ \frac{1}{2}\mathbf{1} \\ \frac{1}{2}\mathbf{1} \\ \mathbf{0} \\ \mathbf{1} \\ \mathbf{0} \\ \mathbf{1} \end{bmatrix} \tag{9.7}$$

$$Z \cup Y = \langle c_u, G_u, G'_u, A_u, A'_u, b_u \rangle \subset \mathbb{R}^N \tag{9.8}$$

3. Abstract Transformers

3.1. General abstract transformer construction.

[3] provide a general method to find sound and minimal area abstract transformers for zonotopes. Sound neuron-wise transformers for the zonotope domaine can be described as:

$$y = \lambda x + \mu + \beta \varepsilon_{\text{new}} \tag{10}$$

For convex C^1 continuous functions, all tangents to the curve of the function yield viable transformers. The resulting parallelogram can be parametrized by the abscissa of the contact point t with $l \leq t \leq u$. Using the mean value theorem and convexity, it follows that there will be a point t_{crit} where the upper edge of the parallelogram will connect the lower and upper endpoints of the graph. For $t < t_{\text{crit}}$ it will make contact on the upper endpoint and for $t > t_{\text{crit}}$ on the lower endpoint. This allows to describe the parameters λ, μ and β of a zonotope transformer for a function $f(x) : \mathbb{R} \to \mathbb{R}$ on the interval $[l, u]$ as:

$$\lambda = f'(t) \tag{11.1}$$

$$\mu = \frac{1}{2}\Big(f(t) - \lambda t + \begin{cases} f(l) - \lambda l, & \text{if } t \geq t_{\text{crit}} \\ f(u) - \lambda u, & \text{if } t < t_{\text{crit}} \end{cases}\Big) \tag{11.2}$$

$$\beta = \frac{1}{2}\Big(\lambda t - f(t) + \begin{cases} f(l) - \lambda l, & \text{if } t \geq t_{\text{crit}} \\ f(u) - \lambda u, & \text{if } t < t_{\text{crit}} \end{cases}\Big) \tag{11.3}$$

$$\nabla_x f(x)|_{x=t_{\text{crit}}} = \frac{f(u) - f(l)}{u - l} \tag{11.4}$$

A minimum area transformer can now be derived by minimizing the looseness μ for $l \leq t \leq t_{\text{crit}}$ and $t_{\text{crit}} \leq t \leq u$. This yields the constrained optimization problems:

$$\min_t \frac{1}{2}(f'(t)(t-u) - f(t) + f(u)), \quad s.t,\ l \leq t \leq t_{\text{crit}} \tag{12.1}$$

$$\min_t \frac{1}{2}(f'(t)(t-l) - f(t) + f(l)), \quad s.t,\ t_{\text{crit}} \leq t \leq u \tag{12.2}$$

These can be solved using the method of Lagrange multipliers. Equation 12.1 leads to the following equations:

$$\mathcal{L} = \frac{1}{2}(f'(t)(t-u) - f(t) + f(u)) + \gamma_1(l-t) + \gamma_2(t-t_{\text{crit}}) \tag{13.1}$$

$$\nabla_t \mathcal{L} = \frac{1}{2}f''(t)(t-u) - \gamma_1 + \gamma_2 = 0 \tag{13.2}$$

$$\nabla_{\gamma_1} \mathcal{L} = t - l \tag{13.3}$$

$$\nabla_{\gamma_2} \mathcal{L} = t - t_{\text{crit}} \tag{13.4}$$

$$\gamma_1 \geq 0 \tag{13.5}$$

$$\gamma_2 \geq 0 \tag{13.6}$$

$$\gamma_1(t-l) = 0 \tag{13.7}$$

$$\gamma_2(t-t_{\text{crit}}) = 0 \tag{13.8}$$

Case 1: Neither constraint is active, $\gamma_1 = \gamma_2 = 0$, $\nabla_t \mathcal{L} = f''(t)(t-u) = 0$. Hence, either $t^* = u = t_{\text{crit}}$, or t^* verifies $f''(t^*) = 0$.

Case 2: $\gamma_1 \neq 0, \gamma_2 = 0$, thus $t^* = l$. In this case, $\gamma_1 = \frac{1}{2}f''(l)(l-u)$. However, as f is convex, $f''(x) \geq 0$, so if $u \neq l$, this leads to $\gamma_1 < 0$ which is not possible.

Case 3: $\gamma_1 = 0, \gamma_2 \neq 0$, thus $t^* = t_{\text{crit}}$ and $\gamma_2 = \frac{1}{2}f''(l)(l-u) \geq 0$.

Case 4: $\gamma_1 \neq 0, \gamma_2 \neq 0$. In this case, $t^* = l = t_{\text{crit}}$.

Analogously, equation Equation 12.1 yields a boundary minimum at $t = t_{\text{crit}}$. Consequently $t = t_{\text{crit}}$ yields the minimum area transformer for convex functions. t_{crit} can be computed either analytically or numerically by solving Equation 11.4 as the point where the local gradient is equal to the mean gradient over the whole interval.

3.2. **Exponential Transformer.**

The exponential function has the feature that its output is always strictly positive, which is important when used as input to the logarithmic function to compute the entropy. Therefore, a guarantee of positivity for the output zonotope is desirable. A constraint yielding such a guarantee can be obtained by inserting $\hat{x}_i = l, \varepsilon_{p+1} = -\operatorname{sign}(\mu)$ and $\hat{y}_i \geq 0$ with $\lambda(t) = e^t$ into Equation 10:

$$0 \leq \lambda l + \frac{1}{2}(f(t) - \lambda t + f(u - \lambda u)) - \frac{1}{2}(\lambda t - f(t) + f(u - \lambda u)) \tag{14.1}$$

$$0 \leq \lambda(l-t) + f(t) \tag{14.2}$$

$$0 \leq e^t(l - t + 1) \tag{14.3}$$

$$t \leq 1 + l \equiv t_{\text{crit},2} \tag{14.4}$$

This constitutes the additional upper limit $t_{\text{crit},2}$ on t. Therefore it is sufficient to reevaluate 16 as it will either be inactive in equation 17 if $t_{\text{crit}} \leq t_{\text{crit},2}$ for the solutions computed previously or the constraints will be insatiable ensuring that 17 will have no solutions. If a strictly positive output is required a small delta can simply be subtracted from the upper limit $t_{\text{crit},2}$. It is easy to see that t is now constrained to $\big[l, \min\big(u, t_{\text{crit},2}\big)\big]$ and that the minimum area solution will be obtained with $t_{\text{opt}} = \min\big(t_{\text{crit}}, t_{\text{crit},2}\big)$. The critical points can be computed explicitly

to $t_{\text{crit}} = \log\left(e^u - e^l\right)$ and $t_{\text{crit},2} = l + 1$. This can be inserted into equations 11 to 14 to obtain a positive, sound and viable transformer.

3.3. **Logarithmic Transformer.**

The logarithmic transformer can be constructed by plugging $f(t) = -\log(t)$ and $f'(t) = -\frac{1}{x}$ into equations 12 to 14 and their results into equation 11. Equation 15 can be solved to $t_{\text{crit}} = \frac{l-u}{\ln(l)-\ln(u)}$.

3.4. **ReLU Abstract Transformer.**

The ReLU abstract transformer defined for the classical Zonotope [4] can be extended naturally to the multi-norm setting [5] since it relies only on the lower and upper bounds of the variables, which are computed using the method described for the Multi-norm Zonotope.

For a zonotope variable x with lower bound l and upper bound u, the Multi-norm Zonotope abstract transformer for $\text{ReLU}(x) = \max(0, x)$ is:

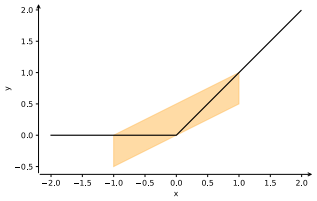
$$y = \begin{cases} 0, & \text{if } u < 0 \\ x, & \text{if } l > 0 \\ \lambda x + \mu + \beta_{\text{new}}\varepsilon_{\text{new}}, & \text{otherwise} \end{cases} \tag{15}$$

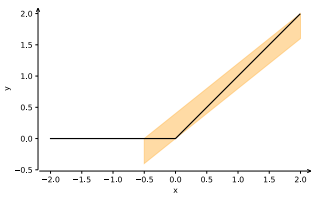
where $\varepsilon_{\text{new}} \in [-1, 1]$ denotes a new noise symbol, and:

$$\lambda = \frac{u}{u - l} \tag{16.1}$$

$$\beta_{\text{new}} = \mu = 0.5 \max(-\lambda l, (1 - \lambda)u) \tag{16.2}$$

We note that the newly introduced noise symbol ε_{new} is an ℓ_∞ noise symbol. This holds for all ε_{new} in the following transformers as well.





3.5. **Tanh Abstract Transformer.**

The abstract transformer for the operation $y = \tanh(x)$ is:

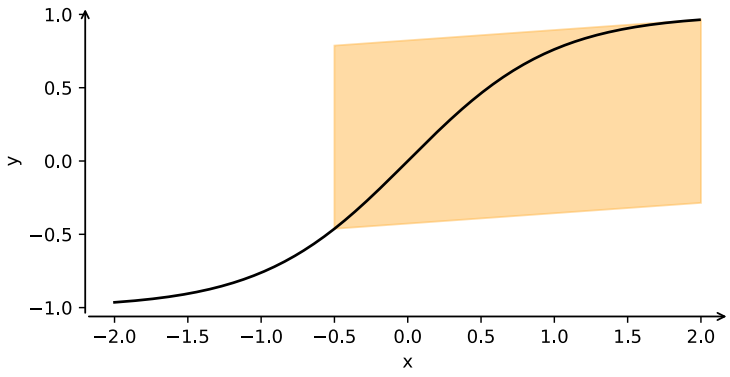
$$y = \lambda x + \mu + \beta_{\text{new}}\varepsilon_{\text{new}} \tag{17}$$

where:

$$\lambda = \min(1 - \tanh^2(l), 1 - \tanh^2(u)) \tag{18.1}$$

$$\mu = \frac{1}{2}(\tanh(u) + \tanh(l) - \lambda(u + l)) \tag{18.2}$$

$$\beta_{\text{new}} = \frac{1}{2}(\tanh(u) - \tanh(l) - \lambda(u - l)) \tag{18.3}$$



3.6. **Exponential Abstract Transformer.**

The operation $y = e^x$ can be modeled through the element-wise abstract transformer:

$$y = \lambda x + \mu + \beta_{\text{new}}\varepsilon_{\text{new}} \tag{19}$$

where:

$$\lambda = e^{t_{\text{opt}}} \tag{20.1}$$

$$\mu = 0.5\big(e^{t_{\text{opt}}} - \lambda t_{\text{opt}} + e^u - \lambda u\big) \tag{20.2}$$

$$\beta_{\text{new}} = 0.5\big(\lambda t_{\text{opt}} - e^{t_{\text{opt}}} + e^u - \lambda u\big) \tag{20.3}$$

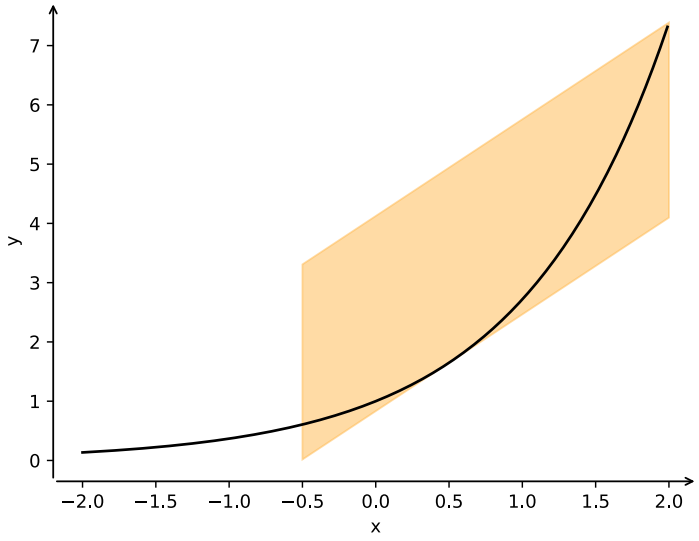
and

$$t_{\text{opt}} = \min\big(t_{\text{crit}}, t_{\text{crit},2}\big) \tag{21.1}$$

$$t_{\text{crit}} = \log\left(\frac{e^u - e^l}{u - l}\right) \tag{21.2}$$

$$t_{\text{crit},2} = l + 1 - \hat{\varepsilon} \tag{21.3}$$

Here, $\hat{\varepsilon}$ is a small positive constant value, such as 0.01. The choice $t_{\text{opt}} = \min\big(t_{\text{crit}}, t_{\text{crit},2}\big)$ ensures that y is positive.



3.7. **Reciprocal Abstract Transformer.**

The abstract transformer for $y = \frac{1}{x}$ with $x > 0$ is given by:

$$y = \lambda x + \mu + \beta_{\text{new}} \varepsilon_{\text{new}} \tag{22}$$

where:

$$\lambda = -\frac{1}{t_{\text{opt}}^2} \tag{23.1}$$

$$\mu = 0.5 \left(\frac{1}{t_{\text{opt}}} - \lambda \cdot t_{\text{opt}} + \frac{1}{l} - \lambda l \right) \tag{23.2}$$

$$\beta_{\text{new}} = 0.5 \left(\lambda \cdot t_{\text{opt}} - \frac{1}{t_{\text{opt}}} + \frac{1}{l} - \lambda l \right) \tag{23.3}$$

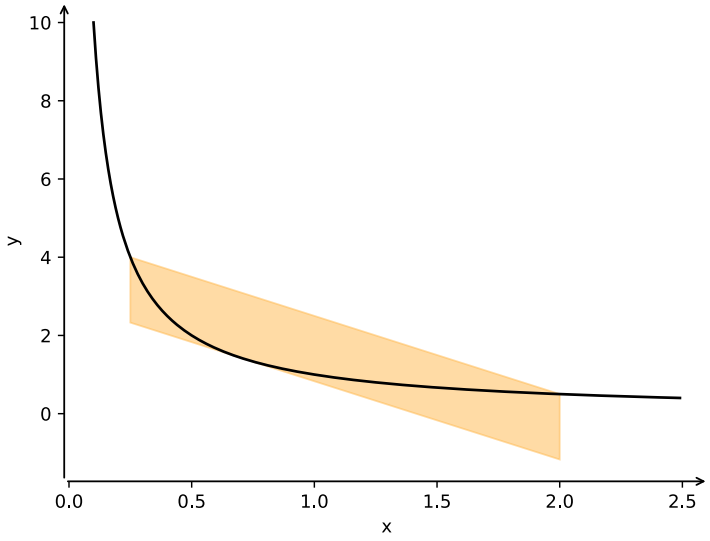
and

$$t_{\text{opt}} = \min\left(t_{\text{crit}}, t_{\text{crit},2}\right) \tag{24.1}$$

$$t_{\text{crit}} = \sqrt{ul} \tag{24.2}$$

$$t_{\text{crit},2} = 0.5u + \hat{\varepsilon} \tag{24.3}$$

Similarly to the exponential transformer, $\hat{\varepsilon}$ is a small positive constant and $t_{\text{opt}} = \min\left(t_{\text{crit}}, t_{\text{crit},2}\right)$ ensures that y is positive.



References

1. Albarghouthi, A.: Introduction to Neural Network Verification. (2021)
2. Bird, T.J.: Hybrid Zonotopes: A Mixed-Integer Set Representation for the Analysis of Hybrid Systems. (2022)
3. Mark Niklas Müller, M.V., Mislav Balunović: Certify or Predict: Boosting Certified Robustness with Compositional Architectures. (2021)
4. Singh, G., Gehr, T., Mirman, M., Püschel, M., Vechev, M.: Fast and effective robustness certification. 10825–10836 (2018)
5. Bonaert, G., Dimitrov, D.I., Baader, M., Vechev, M.: Fast and precise certification of transformers. 466–481 (2021). https://doi.org/10.1145/3453483.3454056

Sécurité des systèmes et des réseaux, Télécom SudParis, Évry
Email address: thomas.winninger@télécom-sudparis.eu
URL: https://sckathach.github.io