RAZeR: Representation Aware Zonotope Reduction

Vladimir Krsmanović
CISPA Helmholtz Center for Information Security
Saarbrücken, Germany
vladimir.krsmanovic@cispa.de

Tobias Ladner
Technical University of Munich
Munich, Germany
tobias.ladner@tum.de

Bernd Finkbeiner
CISPA Helmholtz Center for Information Security
Saarbrücken, Germany
Technical University of Munich
Munich, Germany
finkbeiner@cispa.de

Abstract

Reachability analysis is a popular technique for verifying the safety of a dynamical system under uncertainty, where the uncertainty is often represented by zonotopes. As the size of the zonotope representation typically grows as the reachability analysis progresses, it must be repeatedly reduced at the cost of additional conservatism. While various methods exist to reduce the size of the representation, combined advantages are not fully leveraged as only a single reduction method is typically used throughout the analysis. In this work, we show how dynamically choosing which method to use can yield significantly tighter results, thereby easing verification. This is realized by introducing a heuristic, derived from model predictive control, that accounts for upcoming propagation of the current zonotope. We argue that order reduction is not merely a post-processing step after propagation but also a pre-processing step that shapes how the zonotope evolves under the system dynamics. We evaluate our approach, RAZeR, on nonlinear benchmarks taken from the ARCH-COMP, where we demonstrate that RAZeR produces tighter zonotopes at a negligible performance overhead.

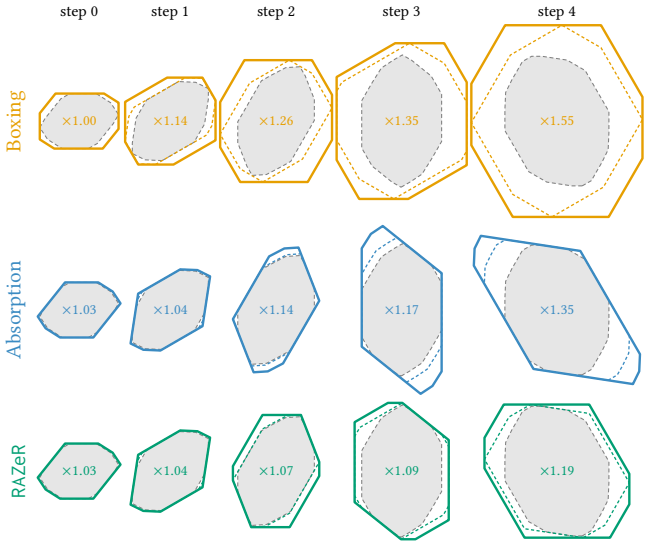


Figure 1: Illustration of how RAZeR improves zonotope reductions in sequential applications by balancing between absorption and boxing of surplus generators in each step.

1 Introduction

A zonotope [8] is a convex, centrally symmetric, continuous set that can be represented with only a few parameters even in high-dimensional spaces. They are a crucial element in reachability analysis of dynamical systems [1, 8], neural network verification [13], and runtime monitoring under uncertainty [5, 6], as the required operations are efficient to compute for zonotopes.

In these applications, the representation of a zonotope usually grows in size, i.e., the number of generators that span the zonotope grows. To remain computationally efficient, zonotope reduction methods try to find another zonotope that encloses the original zonotope with fewer generators, where a more aggressive reduction usually leads to greater conservatism in the overarching goal of verifying a safety-critical system.

Related work. Zonotope reduction methods have long been a subject of study [9]. Common methods simplify the representation of a zonotope by enclosing some generators with an axis-aligned box [3, 8, 9], or absorb them into the remaining generators [12] (Fig. 1). More advanced methods use generator clustering [9] and constrained optimization [11].

These reduction methods are static and focus on computing an enclosure independent of the application. In such applications, a reduction method should not only tightly enclose a given zonotope but also account for its evolution in subsequent steps.

This led to algorithms automatically tuning hyperparameters to balance conservatism across current and future time steps [14]; however, the choice of reduction method was not considered. This choice can be viewed as an optimal control problem over a receding horizon [10]. While this has been shown to be effective, it comes with a higher computational cost and greater system complexity.

Contribution. In this work, we continue this line of research. In summary, our main contributions are:

- We derive a computationally cheap heuristic resembling the future cost of different reduction methods in sequential applications (Sec. 3).
- We use this heuristic to develop our meta-reduction method RAZeR, which dynamically chooses promising reduction methods on the generator level (Sec. 4).
- We evaluate RAZeR on nonlinear benchmarks from ARCH-COMP, resulting in substantially tighter zonotopes at minor runtime overhead (Sec. 5).

2 Background

Notation. We denote vectors with lowercase letters, matrices with uppercase letters, and sets with calligraphic letters. For $n \in \mathbb{N}$, we denote the n-dimensional identity matrix by I_n and use $[n] = \{1, \dots, n\}$. Given a vector $v \in \mathbb{R}^n$, the operation $\mathrm{diag}\,(v)$ creates a matrix with the entries of v on its main diagonal. The concatenation of two matrices $M_1 \in \mathbb{R}^{n \times m_1}$, $M_2 \in \mathbb{R}^{n \times m_2}$ is denoted by $[M_1 M_2] \in \mathbb{R}^{n \times (m_1 + m_2)}$. An interval with bounds $l, u \in \mathbb{R}^n$ is denoted by $[l, u]$, where $l \leq u$ holds element-wise. The bold letter $\mathbf{0}$ denotes a vector with all zeros of proper dimensions.

Set-based computing. Zonotopes are a set representation commonly used in reachability analysis due to their efficient computation of the required operations.

Definition 2.1 (Zonotope [8]). Given a center $c \in \mathbb{R}^n$ and a generator matrix $G \in \mathbb{R}^{n \times h}$, a zonotope is defined by

$$\mathcal{Z} = \langle c, G \rangle_Z = \left\{ c + \sum_{i=1}^{h} \beta_i g_i \;\middle|\; \beta_i \in [-1, 1] \right\}.$$

The most common operations in reachability analysis are the linear map and the Minkowski sum. For a matrix $M \in \mathbb{R}^{m \times n}$, and zonotopes $\mathcal{Z}_1 \langle c_1, G_1 \rangle_Z$, $\mathcal{Z}_2 \langle c_2, G_2 \rangle_Z \subset \mathbb{R}^n$, these are computed by

$$\begin{aligned} M\mathcal{Z}_1 &= \{Mz_1 \mid z_1 \in \mathcal{Z}_1\} = \langle Mc_1, MG_1 \rangle_Z\,, \\ \mathcal{Z}_1 \oplus \mathcal{Z}_2 &= \{z_1 + z_2 \mid z_1 \in \mathcal{Z}_1,\ z_2 \in \mathcal{Z}_2\} = \langle c_1 + c_2, [G_1\ G_2] \rangle_Z\,. \end{aligned} \tag{1}$$

The number of generators h typically grows during reachability analysis, where we use the order of a zonotope $\rho = h/n \geq 0$ as a measure with respect to the system dimension n. To maintain computational efficiency, we have to repeatedly apply an order reduction [9] at the cost of additional outer approximation. For a given zonotope $\mathcal{Z} = \langle c, G \rangle_Z \subset \mathbb{R}^n$,

$$\widehat{\mathcal{Z}} = \mathtt{reduce}\,(\mathcal{Z}, \mathtt{method}, \rho) \supseteq \mathcal{Z} \tag{2}$$

ensures that $\widehat{\mathcal{Z}}$ has at most $n\rho$ generators. There exist various order reduction methods [9], which we broadly categorize into boxing and absorption methods (Fig. 2):

`Girard` [8] splits the generator matrix G into kept generators $G_K \in \mathbb{R}^{n \times (\rho - 1)n}$ and surplus generators $G_S \in \mathbb{R}^{n \times (h - (\rho-1)n)}$, and computes a box enclosure of G_S, resulting in at most n new generators:

$$\widehat{\mathcal{Z}} = \mathtt{reduce}\,(\mathcal{Z}, \mathtt{Girard}, \rho) = \langle c, G_K \rangle_Z \oplus \mathrm{box}\,(\langle \mathbf{0}, G_S \rangle_Z) \supseteq \mathcal{Z}, \tag{3}$$

where

$$\begin{aligned} \mathrm{box}\,(\mathcal{Z}) &= \langle c, \mathrm{diag}\,(\mathrm{rad}(\mathcal{Z})) \rangle_Z \\ \text{with } \mathrm{rad}(\mathcal{Z}) &= \sum_{i=1}^{h} |g_i| \in \mathbb{R}^n. \end{aligned} \tag{4}$$

Similarly, `Combastel` [3] boxes the generators of the smallest Euclidean norm, and `PCA` [9] boxes the surplus in the principal frame of the surplus generators instead of the coordinate frame.

In contrast, `Scott` [12] absorbs surplus generators by scaling the kept generators accordingly. It uses Gauss–Jordan elimination to determine a subset of generators $G^* = [g_1^* \cdots g_n^*] \in \mathbb{R}^{n \times n}$ spanning an invertible basis; the remaining generators are again partitioned

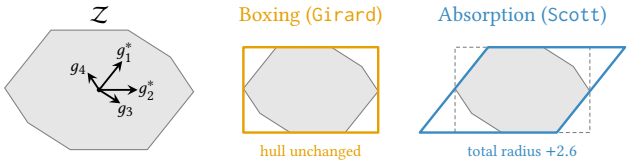


Figure 2: Comparison of boxing and absorption methods

into some generators G'_K that are kept unchanged and the remaining G'_S are subject to the reduction. Without loss of generality, let

$$G = [G^*\ G'_K\ G'_S] = [g_1^* \cdots g_n^*\ \ g_{n+1} \cdots g_{\rho n}\ \ g_{\rho n+1} \cdots g_h]. \tag{5}$$

As G^* forms a basis, we can express G'_S in terms of G^*:

$$V_S = (G^*)^{-1} G'_S, \tag{6}$$

which gets absorbed into the basis by computing:

$$\widehat{G}^* \leftarrow G^* \left(I_n + \mathrm{diag}\,(\mathrm{rad}(\langle \mathbf{0}, V_S \rangle_Z))\right). \tag{7}$$

This effectively computes a box enclosure in the coordinate frame spanned by G^*. As this box is aligned with G^*, each generator of the box can be summed with the respective generator of G^*. Thus, it holds:

$$\widehat{\mathcal{Z}} = \mathtt{reduce}\,(\mathcal{Z}, \mathtt{Scott}, \rho) = \left\langle c, \left[\widehat{G}^*\ G'_K\right] \right\rangle_Z \supseteq \mathcal{Z}. \tag{8}$$

Reachability analysis. We present an intentionally simplified algorithm to conduct reachability analysis of a nonlinear discrete-time system in Alg. 1. This high-level description is sufficient for this work; we defer interested readers to [1] for more information. The initial set $\mathcal{X}_0 \in \mathbb{R}^n$ is iteratively propagated forward in time, where the nonlinear dynamic $x_{k+1} = f(x_k)$ is linearized at each time point k, yielding a linear propagation map $M \in \mathbb{R}^{n \times n}$. This can also be generalized to continuous time systems, i.e., let $A \in \mathbb{R}^{n \times n}$ be the matrix describing the linearized dynamics, then $M = e^{A_k \Delta t}$. To account for the linearization error, the set is expanded after each propagation using (1), where the Minkowski sum results in additional generators. Thus, we are required to repeatedly reduce the order of $\mathcal{X}_k$ to remain computationally feasible.

Algorithm 1: Simplified reachability analysis

Input: System dynamics $f(x)$; initial set $\mathcal{X}_0$; max. order ρ; time horizon $k_{\max}$
Output: Reachable set $\mathcal{R}$

for $k \in [k_{\max}]$ **do**
  Determine linearization point x_k^*
  Compute linear propagation map M using x_k^*
  Compute linearization error $\mathcal{E}$
  $\mathcal{X}_k \leftarrow M\mathcal{X}_{k-1} \oplus \mathcal{E}$ ▷ (1)
  $\mathcal{X}_k \leftarrow \mathtt{reduce}\,(\mathcal{X}_k, \mathtt{method}, \rho)$ ▷ (2)
return $\mathcal{R} \leftarrow \bigcup_k \mathcal{X}_k$

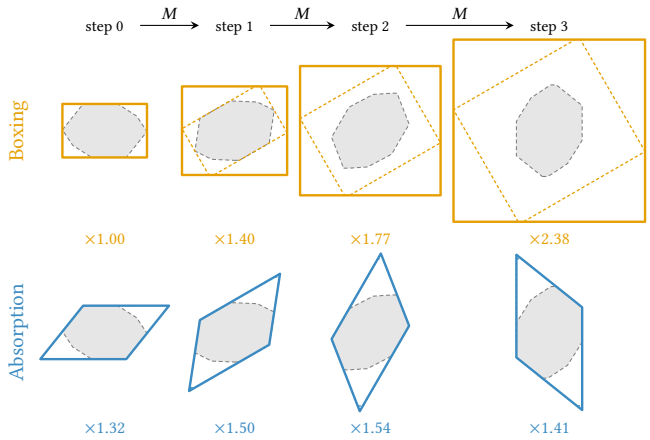


Figure 3: Wrapping effect with an aggressive order reduction to $\rho = 1$ after each step.

3 Cost of Reduction over Time

We measure the cost of the reduction via the volume of the resulting zonotope. As the volume of zonotopes is expensive to compute in higher dimensions [4], we approximate it with the volume of the interval hull as computed in (4). This slightly benefits reduction methods based on boxing; however, as we are considering the entire reachability analysis over time, we assume this effect to be negligible if another method clearly outperforms boxing methods.

In particular, boxing methods are subject to the wrapping effect if applied repeatedly, as illustrated in Fig. 3. Thus, it is reasonable to measure the cost of reduction over time as well. We compute this cost greedily by only looking one time step into the future, effectively bounding the difference of applying (4) before and after this step.

Definition 3.1 (Wrapping cost). For a linear map $M \in \mathbb{R}^{m\times n}$ and a generator $g \in \mathbb{R}^n$, the wrapping cost is given by:

$$\mathrm{W}(M, g) = \||M||g|\|_1 - \|Mg\|_1 \geq 0.$$

The overall cost for applying a boxing method is thus given by:

Definition 3.2 (Boxing cost). For a linear map $M \in \mathbb{R}^{m\times n}$ and a generator $g \in \mathbb{R}^n$, the cost of boxing g is given by:

$$\gamma^{\square}_M(g) = \mathrm{W}(M, g) = \||M||g|\|_1 - \|Mg\|_1 \geq 0.$$

The cost for absorption methods works analogously; we just have to respect the basis G^* while doing so:

Definition 3.3 (Absorption cost). For a linear map $M \in \mathbb{R}^{m\times n}$, a basis $G^* \in \mathbb{R}^{n\times n}$, and a generator $g \in \mathbb{R}^n$ with $v = (G^*)^{-1}g$, the absorption cost of g is given by:

$$\gamma^{\triangle}_M(g) = \mathrm{W}(MG^*, v) = \||MG^*||v|\|_1 - \|Mg\|_1 \geq 0.$$

Using these costs, we provide further insights into how a model predictive control controller [10] chooses which order-reduction method to use. In particular, we notice that `Scott` is preferred as long as the cumulative absorption cost over all generators is below some threshold, and falls back to boxing methods above that threshold. More formally, let the cumulative absorption cost be:

$$\bar{\eta} = \frac{\sum_k \gamma^{\triangle}_{I_n}(g_k)}{\sum_k \|g_k\|_1}, \tag{9}$$

where we used the identity I_n in the numerator, since model predictive control does not directly account for map propagation to future time steps.

Using $\bar{\eta}$, we illustrate this threshold behavior in Fig. 4 by plotting the actions of the controller in an example reachability run with respect to $\bar{\eta}$: Absorption methods work well when the bases are orthogonal to each other, as then the absorption is lossless as it absorbs into a box (7). However, as monitor updates and reductions can introduce shear into the zonotope, this can, over time, lead to greater error. This slowly increases $\bar{\eta}$ until the controller detects the increasing cost on the horizon and switches to boxing methods, which then remove the shear and restore the shape, pushing $\bar{\eta}$ down to levels where absorption again provides more benefit. Across our reachability runs , $\bar{\eta}$ threshold agrees with the chosen method by the controller (`Scott` vs. others) with an AUC of 0.85.

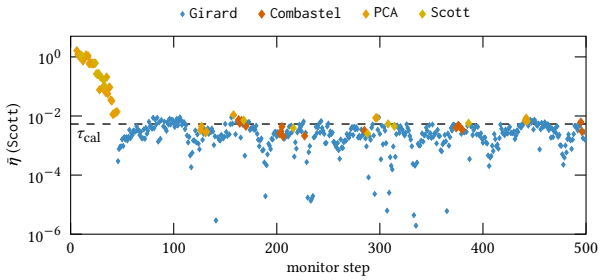


Figure 4: Reduction choices of the Predictive Zonotope Reduction [10] are strongly correlated with the absorption cost.

4 Representation Aware Zonotope Reduction

We build on the observation in Sec. 3 to develop `RAZeR` – standing for Representation Aware Zonotope Reduction – which balances the costs of absorption and boxing for upcoming propagation steps. It uses this information to intelligently select which surplus generators to absorb and which to box, gaining the benefits of model predictive control while avoiding its overhead.

An overview of RAZeR is given in Alg. 2, consisting of two phases: In the *absorption phase*, it absorbs the generators with an expected positive absorption benefit, which can be computed using Def. 3.2 and Def. 3.3:

Definition 4.1 (Absorption benefit). For a linear map $M \in \mathbb{R}^{m\times n}$, a basis $G^* \in \mathbb{R}^{n\times n}$, and a generator $g \in \mathbb{R}^n$ with $v = (G^*)^{-1}g$, the absorption benefit of g is given by:

$$b(v) := b_M(v) = \gamma^{\square}_M(g) - \gamma^{\triangle}_M(g) = \||M||g|\|_1 - \||MG^*||v|\|_1.$$

Afterward, the *boxing phase* encloses all remaining generators using (4) to obtain the desired order.

Please note that the computed absorption benefit depends on the propagation to the next time step, i.e., on the matrix M. For classic

Algorithm 2: Order reduction method RAZeR

Input: Zonotope $\mathcal{Z} = \langle c, G \rangle_Z$; max. order ρ; next map M
Output: $\widehat{\mathcal{Z}} \supseteq \mathcal{Z}$ with at most $n\rho$ generators

1 **if** $h \leq n\rho$ **then return** $\mathcal{Z}$
2 **absorption phase**
3 Determine basis G^* and remainder G' ▷ (5)
4 Compute $b_M(V)$ for $V = (G^*)^{-1}G'$ ▷ *Def. 4.1*, (6)
5 Partition G' into G'_K and G'_S using $b_M(V) > 0$ ▷ (5)
6 $\widehat{G}^* \leftarrow$ Absorb G'_S into basis ▷ (7)
7 **boxing phase**
8 **return** reduce $\left(\left\langle c, \left[\widehat{G}^* \; G_K\right]\right\rangle_Z, \texttt{Girard}, \rho\right)$

reachability analysis, this is computed directly during linearization (Alg. 1, line 3). However, as order reduction is typically applied after the propagation as a post-processing step (line 6), we argue that it should rather be seen as a pre-processing step to prepare a given zonotope for the upcoming propagation.

In settings where we do not have propagation information, such as runtime monitoring under uncertainty [5], we can only estimate the costs of the upcoming propagation: We approximate the absorption benefit using the immediate cost with the identity map, $\gamma^{\triangle}_{I_n}(g)$. However, an equivalent approach to estimating the boxing cost would presume that boxing incurs no wrapping cost, as $\gamma^{\square}_{I_n}(g) = 0$. To be able to account for the error boxing incurs over several reduction steps, we estimate the boxing cost as $\hat{\gamma}^{\square}(g) = \tau\|g\|_1$, where $\tau \geq 0$ controls the preference for absorption and can be calibrated as shown in Fig. 4. Substituting these costs into Def. 4.1 yields

$$b(g) := b_\tau(g) = \hat{\gamma}^{\square}(g) - \gamma^{\triangle}_{I_n}(g) = \tau\|g\|_1 - \gamma^{\triangle}_{I_n}(g). \tag{10}$$

At $\tau = 0$, RAZeR would only absorb when it would be lossless and behave similarly to Girard, while at $\tau \to \infty$, RAZeR would absorb all surplus generators. Crucially, Alg. 2 computes a sound outer approximation independent of the choice of M and τ.

Proposition 4.2 (RAZeR is sound). *Given $\mathcal{Z} = \langle c, G \rangle_Z \subset \mathbb{R}^n$, a maximal order $\rho \geq 1$, and a matrix $M \in \mathbb{R}^{n \times n}$, Alg. 2 computes a zonotope $\widehat{\mathcal{Z}} \supseteq \mathcal{Z}$ with order at most ρ.*

Proof. The outer approximation is given by replicating the absorption of Scott for the generators with a positive absorption benefit, and by the box enclosure of Girard for the remaining generators. The order is ensured by the application of Girard on the zonotope with the absorbed generators (line 8). Thus, the choice of M and τ only affect tightness. □

We can further show that RAZeR 's ability to combine absorption and boxing of surplus vectors yields a better solution than if we treated all generators in the same manner.

Proposition 4.3 (Mixing is better). *Let G^* be a fixed invertible basis and M a linear map. For the surplus generators g_i of G_S, let $v_i = (G^*)^{-1}g_i$. Then, selecting absorption when $b_M(v_i) > 0$ and boxing otherwise yields*

$$\sum_{g_i} \min\{\gamma^{\square}_M(g_i), \gamma^{\triangle}_M(g_i)\} \leq \min\left\{\sum_{g_i} \gamma^{\square}_M(g_i), \sum_{g_i} \gamma^{\triangle}_M(g_i)\right\}. \tag{11}$$

Proof. By Def. 4.1, $b_M(v_i) = \gamma^{\square}_M(g_i) - \gamma^{\triangle}_M(g_i)$, so the selected operation incurs $\min\{\gamma^{\square}_M(g_i), \gamma^{\triangle}_M(g_i)\}$. The claim follows by triangle inequality. □

5 Evaluation

We evaluate RAZeR on eleven scenarios from nonlinear reachability from ARCH-COMP [7] and previous literature on adaptive reachability [15]. All experiments are run in MATLAB R2026a with CORA v2026.1.0 [2] on a laptop with an Intel Core Ultra 7 155U and 32 GB of RAM under Fedora Linux 44.

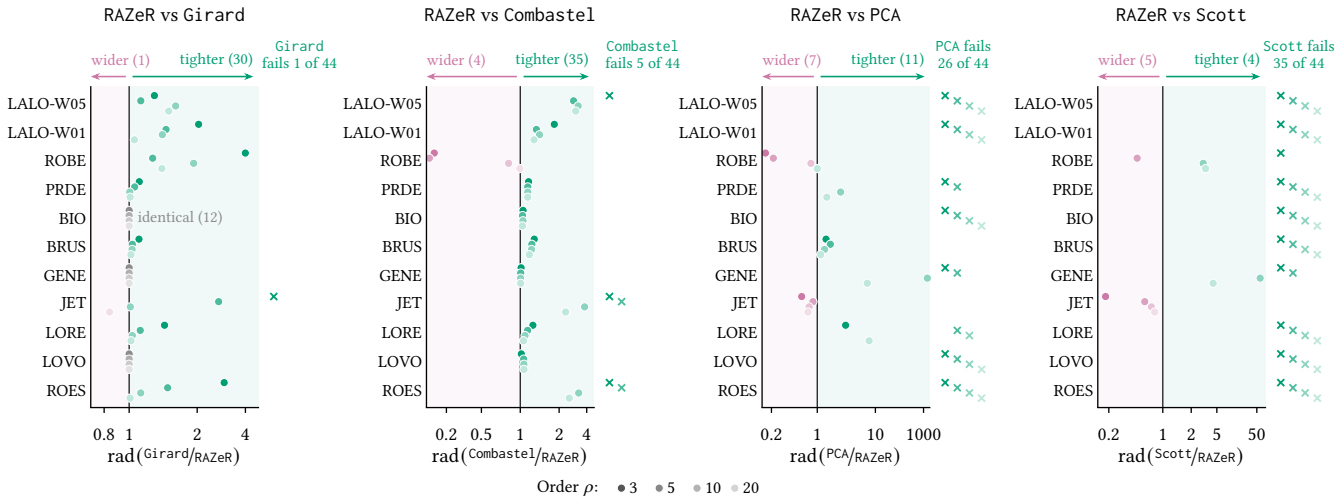


Figure 5: Comparison of RAZeR against each established method using (12) with different orders ρ, on log scale. A value above 1 means that RAZeR is tighter than the respective method. Runs where the reachability analysis using an established method fails due to a set explosion are shown with a cross (×).

5.1 Main Results

We compare the performance of RAZeR against individual reduction methods from Sec. 2 using orders $\rho \in \{3, 5, 10, 20\}$. For all experiments, we show the relative tightness of RAZeR against the respective method in the final time step computed by:

$$\text{rad}(^{\text{method}}/_{\text{RAZeR}}) = \frac{\|\text{rad}(\mathcal{Z}_{\text{method}})\|_1}{\|\text{rad}(\mathcal{Z}_{\text{RAZeR}})\|_1}. \tag{12}$$

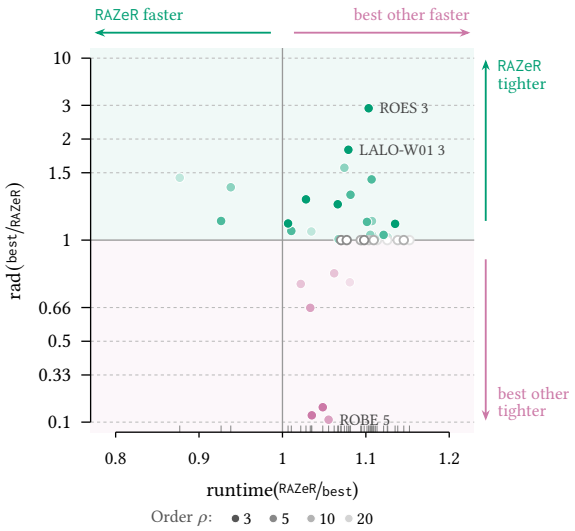


Figure 6: RAZeR compared to the best of Girard, Combastel, Scott and PCA per configuration.

RAZeR obtains substantially tighter results in almost all benchmarks (Fig. 5)—even if it is compared to the best competing method of each benchmark (Fig. 6), which is of course not known in advance. Fig. 6 also shows that the runtime overhead for RAZeR against the competing method is on average only 10%, while automatically tuning the reduction method and thus obtaining tighter results. This can also be seen in Fig. 7, where the radius of RAZeR is within the same order of magnitude as the tightest other method, and even within a factor $\theta \leq 1.5$ in most cases. We have only observed worse behavior on the ROBE and JET benchmarks, which we attribute to system dynamics in these scenarios that lead to inaccurate next-map predictions and, therefore, to suboptimal choices of absorption or boxing, resulting in looser zonotopes.

5.2 Ablation Studies

RAZeR *vs model predictive control.* RAZeR is motivated by model predictive control, but does not come with the computational overhead that usually comes from model predictive control. We adapt the existing model predictive controller for zonotope reduction [10] to better suit the ARCH-COMP setting, and limit it to selecting any of the four reduction methods presented in Sec. 2. We run several variants of these controllers: MPC-G (greedy search, $H = 1$), MPC-B ($H = 3$, $W = 2$), MPC-W ($H = 5$, $W = 4$), with hyperparameter names as defined in [10]. The hyperparameter H effectively counts the number of future time steps considered for the prediction, such

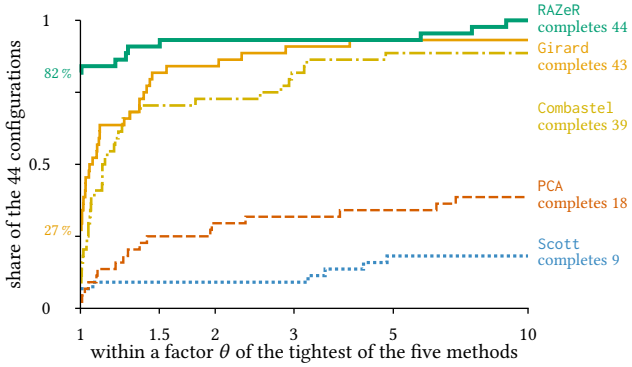


Figure 7: Share of completed runs of each method having its tightness within θ times the radius of the best method.

that we can say that RAZeR uses $H = 1$ as it also greedily looks one time step into the future. However, as shown in Fig. 8, RAZeR matches or even outperforms these model predictive controllers in many scenarios at a fraction of the computational cost; e.g., it is, on average, 5× faster to compute than the next-fastest controller, MPC-G, and 100× faster than MPC-W.

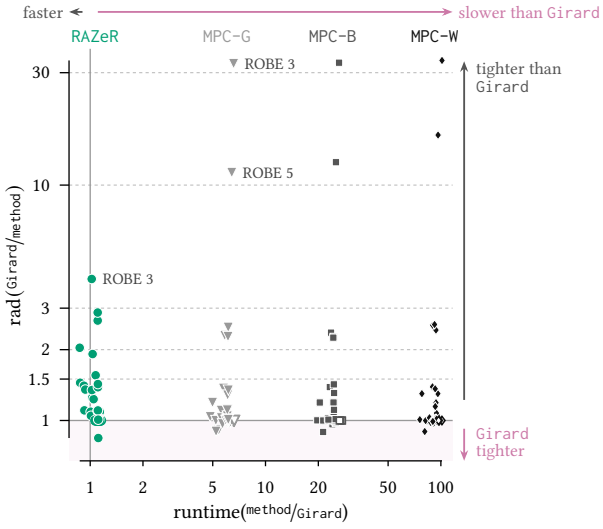


Figure 8: Comparison of RAZeR and the model predictive controllers, both normalized to Girard.

Impact of τ. The threshold parameter τ controls the inflation of absorbing methods (10), and we perform a sweep of its values in Fig. 9. While too much absorption ($\tau \geq 2$) has a negative influence, as we have also seen with Scott above, we found that a moderate balance ($\tau = 0.1$) between the absorption and boxing obtains the best result.

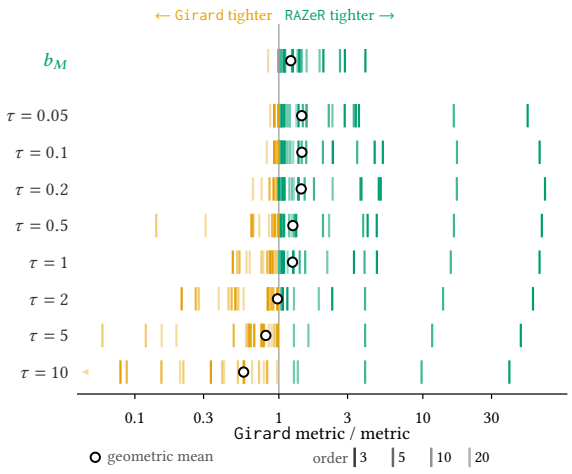


Figure 9: Ablation on absorption threshold τ.

Granularity vs. coarseness. RAZeR operates at the generator level, i.e., it decides for each generator whether it should be absorbed or boxed. To determine if this granularity is necessary, we compare it to a variant, RAZeR-F, which decides on the reduction for the full zonotope using the cumulative absorption benefit (Def. 4.1). The results in Fig. 10 show that RAZeR yields tighter zonotopes compared with RAZeR-F for most benchmarks, while RAZeR-F is substantially tighter for the two ROBE benchmarks.

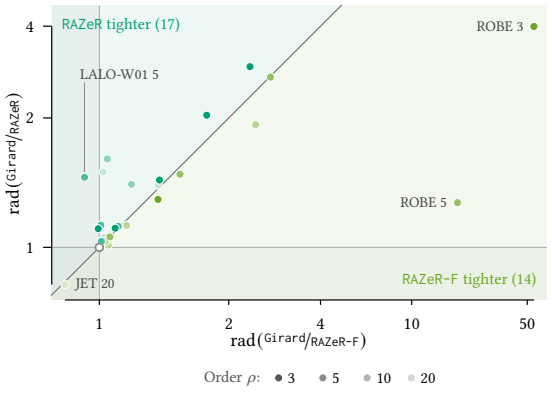


Figure 10: Deciding per generator (RAZeR) or per matrix (RAZeR-F), both against Girard.

6 Limitations

RAZeR is well-suited to sequential zonotope-reduction applications such as reachability analysis, runtime monitoring under uncertainty, or neural network verification. Its effectiveness depends on the linearization of the system dynamics model, or, when this is unavailable, on the choice of a heuristic threshold that must be tuned. In static settings, where there is no temporal propagation of zonotopes, its advantage is limited and comparable to that of existing methods. However, we can push the outer approximation induced by the reduction away from an unsafe specification boundary using RAZeR: For example, let the specification be given by a polytope $\mathcal{P} = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ for some $A \in \mathbb{R}^{p \times n}$ and $b \in \mathbb{R}^p$; then, choosing $M = A$ in Def. 4.1 reduces the outer approximation towards the decision boundary as boxing error in the directions specified in A is minimized. While RAZeR performs well across the ARCH-COMP settings, it does not uniformly outperform all existing methods and incurs additional computational overhead.

7 Conclusion

We presented RAZeR, a new zonotope reduction method that dynamically balances generator absorption and boxing based on the properties of the given zonotope and its upcoming propagation. We argue that zonotope reductions are not just a post-processing step, but also a pre-processing step that shapes how the zonotope evolves with the system dynamics. We use this perspective to derive a computationally cheap heuristic operating at the generator level, thereby deciding whether to box or absorb each surplus generator. This approach combines the complementary strengths of existing reduction methods without requiring an explicit predictive optimization loop. Our evaluation of nonlinear reachability benchmarks demonstrates that RAZeR produces tighter enclosures than established reduction methods in most benchmarks, while preserving sound outer-approximative guarantees. RAZeR matches or outperforms the results of a portfolio of existing reduction methods in 82% of cases, with only moderate runtime overhead.

References

[1] Matthias Althoff. 2010. *Reachability analysis and its application to the safety assessment of autonomous cars.* Ph. D. Dissertation. Technische Universität München.

[2] Matthias Althoff, Niklas Kochdumper, Tobias Ladner, Maximilian Perschl, and Mark Wetzlinger. 2025. CORA manual. *Technical University of Munich* (2025).

[3] Christophe Combastel. 2003. A state bounding observer based on zonotopes. In *2003 European Control Conference.*

[4] Martin Dyer, Peter Gritzmann, and Alexander Hufnagel. 1998. On the complexity of computing mixed volumes. *SIAM J. Comput.* (1998).

[5] Bernd Finkbeiner, Martin Fränzle, Florian Kohn, and Paul Kröger. 2024. Stream-Based Monitoring Under Measurement Noise. In *International Conference on Runtime Verification.*

[6] Bernd Finkbeiner, Martin Fränzle, Florian Kohn, and Paul Kröger. 2026. Cutting Corners on Uncertainty: Zonotope Abstractions for Stream-based Runtime Monitoring. In *International Conference on Runtime Verification.*

[7] Luca Geretti, Julien Alexandre Dit Sandretto, Matthias Althoff, and et al. 2026. ARCH-COMP26 Category Report: Continuous and Hybrid Systems with Nonlinear Dynamics. In *Workshop on Applied Verification for Continuous and Hybrid Systems.*

[8] Antoine Girard. 2005. Reachability of uncertain linear systems using zonotopes. In *International Conference on Hybrid Systems: Computation and Control.*

[9] Anna-Kathrin Kopetzki, Bastian Schürmann, and Matthias Althoff. 2017. Methods for order reduction of zonotopes. In *IEEE Annual Conference on Decision and Control.*

[10] Vladimir Krsmanovic, Florian Kohn, Bernd Finkbeiner, and Milan Simovic. 2026. Predictive Zonotope Reduction: Precise Runtime Monitoring under Uncertainty. *arXiv:2609.03699.*

[11] Sadra Sadraddini and Russ Tedrake. 2019. Linear encodings for polytope containment problems. In *IEEE Annual Conference on Decision and Control.*

[12] Joseph K. Scott, Davide Martino Raimondo, Giuseppe Roberto Marseglia, and Richard D. Braatz. 2016. Constrained zonotopes: A new tool for set-based estimation and fault detection. *Automatica.*

[13] Gagandeep Singh, Timon Gehr, Matthew Mirman, Markus Püschel, and Martin Vechev. 2018. Fast and effective robustness certification. *NeurIPS* (2018).

[14] Mark Wetzlinger, Niklas Kochdumper, and Matthias Althoff. 2020. Adaptive parameter tuning for reachability analysis of linear systems. In *IEEE Annual Conference on Decision and Control.*

[15] Mark Wetzlinger, Adrian Kulmburg, Alexis Le Penven, and Matthias Althoff. 2022. Adaptive reachability algorithms for nonlinear systems using abstraction error analysis. *Nonlinear Analysis: Hybrid Systems.*

Acknowledgements

This work was partially supported by the European Research Council (ERC) Grant HYPER (No. 101055412). Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. This work was also supported by the project AL 1185/33-1 funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG).

A Detailed Evaluation Results

Detailed evaluation results on each benchmark and order ρ are given in Tab. 1 for our evaluation metric (12), and in Tab. 2 for the computation time.

Table 1: Absolute final metric values at fixed zonotope order ρ; smaller is tighter, and bold marks the smallest value of a row at the precision shown.

								RAZeR, threshold τ							
Scenario	Metric	ρ	Girard	Combastel	PCA	Scott	RAZeR	0.05	0.1	0.2	0.5	1	2	5	10
LALO-W05	x_4	3	1.69	F	F	F	1.35	0.469	**0.32**	0.342	F	F	F	F	F
		5	0.203	0.515	F	F	0.183	**0.0913**	0.1	0.134	F	F	F	F	F
		10	0.0924	0.188	F	F	**0.0593**	0.0599	0.0811	0.0908	0.656	F	F	F	F
		20	0.0605	0.124	F	F	**0.0418**	0.0508	0.062	0.0915	0.195	F	F	F	F
LALO-W01	x_4	3	0.0255	0.0226	F	F	0.0125	7.46×10^{-3}	5.51×10^{-3}	$\mathbf{5.01\times10^{-3}}$	6.17×10^{-3}	7.68×10^{-3}	0.0109	0.0408	F
		5	6.48×10^{-3}	5.94×10^{-3}	F	F	4.61×10^{-3}	$\mathbf{4.15\times10^{-3}}$	4.18×10^{-3}	4.71×10^{-3}	5.03×10^{-3}	4.64×10^{-3}	6.54×10^{-3}	0.0133	F
		10	4.38×10^{-3}	4.41×10^{-3}	F	F	$\mathbf{3.24\times10^{-3}}$	3.85×10^{-3}	4.00×10^{-3}	4.13×10^{-3}	4.88×10^{-3}	5.79×10^{-3}	9.82×10^{-3}	0.0368	F
		20	3.25×10^{-3}	3.86×10^{-3}	F	F	$\mathbf{3.11\times10^{-3}}$	3.74×10^{-3}	3.95×10^{-3}	4.29×10^{-3}	4.44×10^{-3}	5.46×10^{-3}	6.53×10^{-3}	0.0167	0.419
ROBE	$2\,\omega$	3	0.0748	2.50×10^{-3}	2.73×10^{-3}	F	0.0188	1.40×10^{-3}	1.16×10^{-3}	$\mathbf{1.06\times10^{-3}}$	1.12×10^{-3}	1.16×10^{-3}	1.29×10^{-3}	1.56×10^{-3}	1.88×10^{-3}
		5	0.0175	1.59×10^{-3}	3.13×10^{-3}	7.71×10^{-3}	0.0142	1.07×10^{-3}	$\mathbf{1.02\times10^{-3}}$	$\mathbf{1.02\times10^{-3}}$	1.07×10^{-3}	1.12×10^{-3}	1.27×10^{-3}	1.51×10^{-3}	1.79×10^{-3}
		10	2.36×10^{-3}	1.03×10^{-3}	1.05×10^{-3}	3.63×10^{-3}	1.23×10^{-3}	1.00×10^{-3}	$\mathbf{9.98\times10^{-4}}$	$\mathbf{9.98\times10^{-4}}$	1.06×10^{-3}	1.09×10^{-3}	1.25×10^{-3}	1.47×10^{-3}	1.74×10^{-3}
		20	1.26×10^{-3}	$\mathbf{9.32\times10^{-4}}$	9.40×10^{-4}	3.00×10^{-3}	9.37×10^{-4}	9.55×10^{-4}	1.00×10^{-3}	9.93×10^{-4}	1.03×10^{-3}	1.10×10^{-3}	1.21×10^{-3}	1.42×10^{-3}	1.61×10^{-3}
PRDE	x_3	3	0.609	0.634	F	F	0.559	0.559	0.557	**0.556**	0.56	0.561	0.572	0.62	F
		5	0.582	0.622	F	F	0.556	0.558	0.556	**0.554**	**0.554**	0.561	0.559	0.599	0.702
		10	0.556	0.619	1.08	F	0.553	**0.552**	**0.552**	**0.552**	0.553	0.553	0.566	0.599	1.07
		20	0.554	0.616	0.702	F	**0.55**	**0.55**	**0.55**	**0.55**	0.552	0.555	0.564	0.591	0.673
BIO	x_1	3	0.0187	0.0194	F	F	0.0187	0.0188	**0.0186**	0.019	0.0199	0.0206	0.0374	F	F
		5	0.0177	0.0183	F	F	0.0177	0.0176	**0.0174**	0.0181	0.0197	0.0213	0.0377	F	F
		10	**0.0172**	0.0179	F	F	**0.0172**	**0.0172**	0.0174	0.0179	0.0188	0.0194	0.0329	0.289	F
		20	0.0171	0.0176	F	F	0.0171	**0.017**	0.0172	0.018	0.0186	0.0192	0.03	0.111	F
BRUS	x_1	3	0.0995	0.114	0.114	F	**0.0915**	0.095	0.0942	0.0934	0.0959	0.0948	0.102	0.147	0.295
		5	0.0913	0.106	0.125	F	0.0889	0.0916	0.0897	**0.0888**	0.0893	0.0911	0.103	0.148	1.04
		10	0.0883	0.102	0.103	F	0.0861	**0.086**	0.0868	0.0866	0.0879	0.0917	0.103	0.148	0.218
		20	0.0868	0.0982	0.0929	F	**0.0856**	0.0865	0.086	0.086	0.0875	0.0912	0.102	0.139	0.209
GENE	x_1	3	**0.111**	0.113	F	F	**0.111**	**0.111**	**0.111**	**0.111**	0.112	0.129	0.133	0.142	0.195
		5	0.11	0.111	F	F	0.11	0.11	0.111	0.11	**0.108**	0.126	0.131	0.143	0.191
		10	**0.108**	0.109	1.87×10^{2}	7.01	**0.108**	**0.108**	**0.108**	0.109	**0.108**	0.123	0.129	0.136	0.148
		20	**0.107**	0.108	0.667	0.459	**0.107**	**0.107**	0.108	0.108	**0.107**	0.112	0.116	0.125	0.164
JET	x_1	3	F	F	0.105	**0.0276**	0.159	0.0489	0.0433	0.0399	0.0305	0.0286	0.0278	0.0277	0.0277
		5	0.11	F	0.0373	**0.0275**	0.0414	0.0332	0.0314	0.0299	0.0286	0.0279	0.0277	0.0277	0.0277
		10	0.0352	0.131	0.0284	**0.0273**	0.0349	0.0302	0.0293	0.0285	0.0279	0.0277	0.0277	0.0277	0.0277
		20	0.0269	0.0738	**0.0252**	0.0269	0.0319	0.0289	0.0283	0.028	0.0277	0.0277	0.0277	0.0277	0.0277
LORE	x_1	3	0.371	0.326	0.627	F	**0.268**	0.271	0.271	0.272	0.281	0.296	0.324	0.491	4.67
		5	0.292	0.296	F	F	**0.265**	0.267	0.267	0.268	0.274	0.288	0.321	0.455	1.91
		10	0.27	0.283	F	F	**0.263**	0.264	0.265	0.267	0.272	0.285	0.317	0.442	1.25
		20	0.267	0.276	1.81	F	**0.263**	0.264	0.264	0.266	0.271	0.284	0.319	0.449	1.3
LOVO	x_1	3	**0.138**	0.14	F	F	**0.138**	0.149	0.149	0.151	0.214	0.289	0.655	F	F
		5	**0.134**	0.14	F	F	**0.134**	0.143	0.14	0.155	0.201	0.251	0.505	F	F
		10	**0.133**	0.14	F	F	**0.133**	**0.133**	**0.133**	**0.133**	0.145	0.259	0.479	F	F
		20	0.133	0.14	F	F	0.133	**0.132**	0.133	0.134	0.156	0.212	0.348	F	F
ROES	x_1	3	24.8	F	F	F	8.63	8.65	10.5	6.62	5.18	**5.17**	F	F	F
		5	7.93	F	F	F	5.55	5.4	5.51	4.52	**3.92**	5.2	F	F	F
		10	4.53	13.1	F	F	4.1	**3.81**	3.83	3.82	3.85	5.05	F	F	F
		20	3.59	8.97	F	F	3.56	**3.48**	3.49	3.55	3.69	4.61	F	F	F
Completed			43/44	39/44	18/44	9/44	44/44	44/44	44/44	44/44	42/44	40/44	36/44	30/44	24/44

Table 2: Median reachability times in seconds over three measured repetitions, for the configurations and columns of Tab. 1; bold marks the fastest completed run of a row at the precision shown. Diagnostics and trace extraction are excluded.

							RAZeR, threshold τ							
Scenario	ρ	Girard	Combastel	PCA	Scott	RAZeR	0.05	0.1	0.2	0.5	1	2	5	10
LALO-W05	3	**38.4**	F	F	F	39.5	47.2	43.1	48.9	F	F	F	F	F
	5	38.6	**35.2**	F	F	35.7	43.5	43.7	76.1	F	F	F	F	F
	10	**34.3**	**34.3**	F	F	36.9	43.6	43.6	77.7	79.5	F	F	F	F
	20	42.9	**34.2**	F	F	37.6	46.2	46	42.8	42.7	F	F	F	F
LALO-W01	3	41.7	**33.7**	F	F	36.4	43.7	42.5	40.6	40.2	40.3	40.1	50	F
	5	39.4	**33.6**	F	F	36.3	46.3	46.9	40.5	40.8	40.7	40.8	45.5	F
	10	39.3	**33.9**	F	F	36.8	47.8	44.7	41	41.4	42	41.5	48.1	F
	20	36.5	**34.1**	F	F	37.7	47.8	46	43.3	43.2	43	43.2	49.4	48.9
ROBE	3	66	**65.1**	66.6	F	67.3	81.1	74.7	72.6	73.5	74.1	74.5	84.6	83.4
	5	65.2	**64.7**	72.4	70.9	68.2	75.6	76.6	87.5	75.1	75.4	75.6	86	84.5
	10	66.7	**64.8**	68.2	74.6	68.9	79.1	84	76.3	78	78.6	77.9	88.8	87.2
	20	67.5	**64.6**	68.1	78.2	69.1	87.3	84.3	81.6	84	84	84.6	95.1	93.2
PRDE	3	78.8	**75.3**	F	F	79.3	90.3	88.5	85	87.4	87.5	91.8	99.3	F
	5	77.8	**75.8**	F	F	78.7	85.9	90.3	89.3	86.8	222	223	98.9	97.1
	10	**74.3**	74.9	80.7	F	79.3	91.9	91.4	232	244	245	240	100	102
	20	**71.3**	75.2	73.6	F	79.1	98.9	98.6	256	253	253	253	102	102
BIO	3	**7.58**	8.03	F	F	8.11	9.32	9.27	23.4	25	26.1	9.6	F	F
	5	**7.37**	7.84	F	F	8.44	9.04	9.15	9.8	9.24	9.37	9.39	F	F
	10	**7.3**	7.84	F	F	8.31	9.34	9.42	9.42	9.65	9.91	9.54	11.1	F
	20	**7.31**	7.93	F	F	8.42	9.33	9.48	9.52	10	10.3	10.5	11.6	F
BRUS	3	**14.3**	15.5	14.6	F	16.3	17.5	17.6	17.3	17	17.1	19.2	20.5	19.9
	5	**14.5**	15.2	14.8	F	16.3	17.6	17.7	18.2	17.6	17.6	18	20	19.9
	10	**14.7**	15.4	14.9	F	16.3	17.7	17.7	18.4	18.8	18.3	19	20.2	20.3
	20	**14.6**	15.3	14.9	F	16.5	17.8	17.8	18	18.1	18.8	18.5	20.4	20.6
GENE	3	**4.03**	4.26	F	F	4.43	4.96	5.05	5.1	5.72	5.51	5.29	6.03	5.96
	5	**4.2**	4.21	F	F	4.66	5.11	5.09	5.27	5.36	5.35	6	6.24	6.12
	10	4.23	**4.18**	4.87	6.19	4.63	5.23	5.17	5.31	5.45	5.85	5.88	6.41	6.45
	20	**4.29**	4.42	4.44	6.88	4.83	5.44	5.36	5.63	5.62	5.67	6.08	6.85	6.91
JET	3	F	F	**23**	24.2	25.4	27.7	27.7	27.8	27.4	27.4	27.7	31.6	31.3
	5	**22.6**	F	23	24.1	24.9	27.5	27.7	30.8	30.5	30.2	29.5	34	31
	10	**22.7**	23.6	23.1	24.6	25.1	27.6	28	28	28.1	29.1	73	35	31.5
	20	**22.6**	23.6	23.3	25.4	25.1	28.2	28.5	73.9	77.5	76	76.6	37.1	33.2
LORE	3	**29.8**	31	30.6	F	33.1	36.2	35.7	93.4	93	93.2	94.4	50	40.9
	5	**30**	30.9	F	F	33.1	35.9	36.3	93.6	94.1	35.2	35.4	48.9	40.9
	10	**30.1**	31.2	F	F	33.3	36.6	36.3	35.2	35.1	35.4	35.8	53.9	45.2
	20	**30.1**	30.9	30.9	F	33.3	37	37.4	35.7	36.2	36.1	36.5	59.9	46.8
LOVO	3	**19.7**	20	F	F	21.2	23	23	22.1	22.1	22.2	22.3	F	F
	5	**19.3**	20.1	F	F	21.2	23.3	23.4	22.2	22.2	22.2	22.5	F	F
	10	**19.4**	20	F	F	21.4	23	25.7	22.4	22.3	22.6	23	F	F
	20	**19.5**	19.9	F	F	21.7	24.6	22.8	22.7	23	23.1	23.7	F	F
ROES	3	**17.8**	F	F	F	19.7	22.3	21.5	21	20.9	21.1	F	F	F
	5	**17.6**	F	F	F	19.5	22.9	21.7	21	21.2	21.4	F	F	F
	10	**17.6**	18.3	F	F	19.5	22.6	23.9	21.1	21.5	21.7	F	F	F
	20	**17.7**	18.4	F	F	19.7	22.5	25.1	21.5	22	23	F	F	F
Completed		43/44	39/44	18/44	9/44	44/44	44/44	44/44	44/44	42/44	40/44	36/44	30/44	24/44