

Contents

1 Changelog	1
1.1 General	1
1.2 Abstract (Re-read required later on)	2
1.3 Introduction	2
1.4 2.1 Encoding ranges as bounds	2
1.5 2.2 128 bit compressed bounds	2
1.6 2.3 Instrumenting block based allocator	3
1.7 Memory allocator design	3
1.8 Evaluation	3
1.9 Related work	4
1.10 Conclusion	4

1 Changelog

1.1 General

- Change name (Thinking of using (Fat Address Translation) -> similar to GNU))
- Avoid words which create options
- Use citation with “~/cite{foo}”
- Clear distinction on Literature review and contribution done
- Generalise capitalisation
- Instead of "Leverage" -> "use"
- ☐ Remove whitespace from figures to be compressed
- ☐ Uniform font sizes in figures
- ☐ Symmetrical lines on the figure
- TLB with acronym only once for the first instance in the paper

1.2 Abstract (Re-read required later on)

- Reduce Abstract to a single paragraph'
 - "Mitigate gap, capacity" -> Expand on for instance on how is it measured.
 - Huge pages moved to the introduction.
- Robs proposed structure (This paper presents <NAME> user-space memory allocator. It uses capability-based addressing to <list properties>).

1.3 Introduction

- "It becomes possible" expand more with baseline compressed bounds (Relook into this later on).
 - "This paper makes the following contribution" -. bullet points with marked sections.
 - Research questions removed.
 - "This implementation" Specify implementation
- Robs recommended Introduction structure (Expansion of the abstract, Name of the existing system, clearly describe limitation, Introduce memory allocator).

1.4 2.1 Encoding ranges as bounds

- Repetition "Tracking of memory ranges using information in pointers" -> remove this
- Paraphrase last sentence (To be looked into later on again).

1.5 2.2 128 bit compressed bounds

- Cites Jemalloc paragraph.
- Last sentence needs rewriting. (Needs re-check)

1.6 2.3 Instrumenting block based allocator

- Needs citation "Which increases overhead and adds complexity"
- Elaborate on "this method"
- "precise" memory management (Elaborate on this and prove which aspects of the other allocator are not precise). (Todo be looked later on).

1.7 Memory allocator design

- Remove from heading "sample" and ":"
- Remove future work paragraph
- "unmapped" -> unmap
- "Risks of memory leaks with metadata lookups or complex data structures" -> Elaborate on this (Commented until further notice).

1.8 Evaluation

- Name of the project keeps the sentence shorter.
- "Such as" -> Definitive and exhaustive.
- Section reference on evaluation expanded.
- Remove "Providing a solid foundation for meaningful comparisons"
- Decide to use Macro or instead "real world C programs"
- Experiment section typo heading
- Fix Latex benchmarks structure
- Change from "Diagram" to "Figure"
- Result para 2 -> too wordy
- "aimed" first use of past tense (Recommendation: avoid)
- Rewrite usability (Changed to Analysis)

1.9 Related work

- Related work lifted up after introduction based on the reflection of the feedback.

1.10 Conclusion

- Rewrite