# CompositionMeasure for MaxDivergence

## Michael Shoemate

Proves soundness of the implementation of `CompositionMeasure` for `MaxDivergence` in `mod.rs` at commit f5bb719 (outdated[1]).

# 1 Hoare Triple

## Precondition

### Compiler-Verified

Types matching pseudocode.

### Caller-Verified

None

## Pseudocode

```
1  class CompositionMeasure(MaxDivergence):
2      def composability(   #
3          self, adaptivity: Adaptivity
4      ) -> Composability:
5          if matches(adaptivity, Adaptivity.FullyAdaptive):
6              raise "fully-adaptive composition is not currently supported for max-divergence"
7          return Composability.Concurrent
8
9      def compose(self, d_mids: Vec[Self_Distance]) -> Self_Distance:
10         d_out = 0.0
11         for d_mid in d_mids:
12             d_out = d_out.inf_add(d_mid)
13         return d_out
```

## Postcondition

**Theorem 1.1.** `composability` returns `Ok(out)` if the composition of a vector of privacy parameters `d_mids` is bounded above by `self.compose(d_mids)` under `adaptivity` adaptivity and `out`-composability. Otherwise returns an error.

*Proof.* By the postcondition of `InfAdd` we have that $\sum_i \texttt{d\_mids}_i \leq \texttt{compose(d\_mids)}$.

| Adaptivity | Sequential | Concurrent |
|---|---|---|
| Non-Adaptive | Theorem 1[DMNS06] | Theorem 1.8[VW21] |
| Adaptive | Theorem 1[DMNS06] | Theorem 1.8[VW21] |
| Fully-Adaptive | None | None |

---

[1]See new changes with `git diff f5bb719..5114655 rust/src/combinators/sequential_composition/mod.rs`

This table is reflected in the implementation of `composability` on line 2.

$\square$

# References

[DMNS06]  Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography*, pages 265–284, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[VW21]     Salil Vadhan and Tianhao Wang. Concurrent composition of differential privacy, 2021.