fn make_fully_adaptive_composition

Michael Shoemate

> This proof resides in **"contrib"** because it has not completed the vetting process.

Proves soundness of `fn make_fully_adaptive_composition`.

1 Hoare Triple

Precondition

Compiler-verified

- Argument `input_domain` of type `DI`.
- Argument `input_metric` of type `MI`.
- Argument `output_measure` of type `MO`.
- Generic `DI` implements `Domain`.
- Generic `MI` implements `Metric`.
- Generic `MO` implements `Measure`.
- `(DI, MI)` implements `MetricSpace`.

User-verified

Pseudocode

```
def make_sequential_odometer(
    input_domain: DI,
    input_metric: MI,
    output_measure: MO,
) -> Odometer[DI, MI, MO, Measurement[DI, TO, MI, MO], TO]:
    def function(arg: DI_Carrier, wrapper: Wrapper | None):
        return new_sequential_odometer_queryable(
            input_domain,
            input_metric,
            output_measure,
            arg,
            wrapper)

    return Odometer.new(
        input_domain,
        Function.new_interactive(function),
        input_metric,
        output_measure)
```

Postcondition

For every setting of the input parameters `(input_domain, input_metric, output_measure, DI, TO, MI, MO)` to `make_fully_adaptive_composition` such that the given preconditions hold, `make_fully_adaptive_compositior` raises an exception (at compile time or run time) or returns a valid odometer. A valid odometer has the following properties:

1. (Data-independent exceptions). For every pair of elements x, x' in `input_domain`, `function`(x) and `function`(x') either both raise an exception, or neither raise an exception.

2. (Wrapping guarantee). Interactive measurement queryables spawned while evaluating external queries are wrapped by the wrapper function accompanying the external query.

3. (Valid odometer queryable). For every element x in `input_domain`, where `function`(x) does not raise an exception, `function`(x) returns a valid odometer queryable.

Proof. (Data-independent exceptions). The only function called, `new_sequential_odometer_queryable`, does not raise an exception, as verified by the compiler. Therefore all invocations of `function` do not raise an exception. □

Proof. (Valid odometer queryable). Under the assumption that the input data is a member of the input domain, the precondition of `make_fully_adaptive_composition_queryable` is met, so by its postcondition the return value is a valid odometer queryable. □