A Clearer View on Fairness: Visual and Formal Representations for Comparative Analysis

Julian Alfredo Mendez* 0000−0002−7383−0529 julian.mendez@cs.umu.se
Timotheus Kampik 0000−0002−6458−2252 tkampik@cs.umu.se
Andrea Aler Tubella 0000−0002−8423−8029 andrea.aler@upc.edu
Virginia Dignum 0000−0001−7409−5813 virginia@cs.umu.se

Department of Computing Science, *Umeå University*, Umeå, Sweden

***Abstract*—The opaque nature of machine learning systems has raised concerns about whether these systems can guarantee fairness. Furthermore, ensuring fair decision making requires the consideration of multiple perspectives on fairness. At the moment, there is no agreement on the definitions of fairness, achieving shared interpretations is difficult, and there is no unified formal language to describe them. Current definitions are implicit in the operationalization of systems, making their comparison difficult. In this paper, we propose a framework for specifying formal representations of fairness that allows instantiating, visualizing, and comparing different interpretations of fairness. Our framework provides a meta-model for comparative analysis. We present several examples that consider different definitions of fairness, as well as an open-source implementation that uses the object-oriented functional language SODA.**

***Index Terms*—Responsible artificial intelligence · Ethics in artificial intelligence · Formal representation of fairness**

I. Introduction

A key challenge in ensuring or assessing fairness is the heterogeneity of perspectives on fairness, because there is no canonical definition of what is fair and what is not. In particular, fairness is not a "one-size-fits-all"-problem: there is no unique operationalizable definition of fairness. In fact, research in various areas of formal definitions of fairness has increased considerably [15]. In the machine learning community, different frameworks have been presented to quantify fairness in classification [3], [5]. Even if fairness can be seen as "the absence of prejudice or favoritism towards an individual or group based on its inherent or acquired characteristics" [29], different criteria can be used to determine fairness of decisions, and many of them should be specifically formulated to be clear to those involved. Determining what is fair varies between cultures [10], and even within the same culture, different individuals can perceive fairness differently [13].

Agreeing on a particular notion of fairness or facilitating an understanding of the diversity of perspectives on fairness can avoid conflicts. A structured discussion and analysis of fairness requires a framework for specifying and comparing perspectives on fairness to enable the elicitation of differences and ultimately desiderata that stakeholders can agree on. Although agreements on the interpretation of fairness or other societal values are complex, a growing number of approaches are being proposed at both theoretical and practical levels, particularly following the Design for Values methods [35], [16], [36].

This paper uses the ACROCPoLis framework [2], which provides a shared vocabulary for fairness assessments, making explicit the relevant factors and their relations. This allows for comparison of similar situations, highlighting differences in dissimilar situations, and capturing different interpretations by different stakeholders. This framework is the underpinning to obtain an applicable framework for operationalizing fairness by:

i. introducing Tiles (Transparent, Intuitive, Logical, Ethical, and Structured), a visual specification language especially tailored for fairness definitions;
ii. presenting a formal meta-model and examples of fairness definitions using Tiles; and
iii. providing an implementation of Tiles in an object-oriented functional language.

The remaining sections are structured as follows. Section II provides an overview of the state of the art, and in particular of challenges regarding the formalization of fairness. Then, Section III provides an informal conceptualization of fairness (drawing from existing research) and introduces a formal meta-model for fairness, as well as Tiles, the corresponding approach to implementation and visualization of fairness models. Formalization and implementation are illustrated using several simple examples in Section IV. Finally, we conclude the article with a discussion of related work and an outline of future research directions in Section V.

II. Background

While fairness is a crucial societal concept, its definition, even in a specific context, is typically subjective. For example, when a state provides childcare subsidies to a family, a "fair" distribution may be colloquially defined in the following ways, among others:

- per child, every family receives the same amount of subsidies;
- per child, subsidies depend on family income, i.e., the amount of subsidies increases with decreasing income;

* Corresponding author. The authorship order is by relative overall contributions to the manuscript.

- per child, subsidies depend on family income and the number of older siblings, i.e., the amount of subsidies per child increases with an increasing number of children.

Each option may be considered fair; one cannot objectively stipulate that one option is necessarily "fairer" than the other. Different communities may have different opinions about what a fair childcare subsidy is [8]. For example, the province of Manitoba, Canada, considers these relevant factors: family income, number and age of the children, number of days required for care, and reason for care [21]. Similarly, the Australian Government publishes a structure diagram of how some factors weigh on the allocation of the childcare subsidy, especially income [12], as the subsidy rate is lowered, in stages, as family income increases, and reaches zero for families with an annual income of or above 352,453 AUD (in 2019-2020).

Comparing different scenarios is a complex task, especially for those who are not specialized in the topic. Thus, a formal diagram can help visualize the differences between criteria of two different countries, or the same country at different points in time. However, creating a system to design such diagrams is challenging, as informal descriptions carry the risk of inconsistencies and flawed modeling. This risk may be reduced if we are able to categorize the different fairness scenarios and provide pre-built consistent blocks to model them. Each block works as a logical unit that is small enough to be fully understood, but powerful enough to require only a few blocks for a standard diagram.

Two prominent categories of *scenarios* pertaining to fairness are *resource allocation scenarios* and *scoring scenarios*. Given a group of individuals, resource allocation scenarios focus on how to find an optimal allocation of a fixed amount of resources [24]. The value of resources is abstracted by a *utility function*, which is a function that gives a comparable value to resources. The utility function may represent qualities or quantities, such as money, time, weight, and size. Implementing fairness in resource allocation is a challenging task because fairness and efficiency are competing objectives [6]. The Gini index [19], [20] and the points on the Lorenz curve [17], [18] are well-known approaches to fairness in resource allocation scenarios and provide frequently used measures for wealth (in)equality in a macroeconomic context.

Scoring scenarios focus on how fair a scoring of a group of individuals is based on their individual attributes. Individuals receive a score based on their attributes, abstracted by a *scoring function*, which is a function that gives a comparable score to individuals with respect to some aspect. This score may assess the likelihood that an individual is able to repay a loan or is a good fit for a particular job position.

To check whether the scoring function itself is fair with different individuals, we could use a counterfactual check [25], especially considering that protected attributes, such as gender, ethnic origin, social status, age, and sexual orientation, can be "noisy", and produce unfair scoring [30]. However, removing or exchanging protected attributes could have limitations, as attributes often contain confounding factors and correlations that are difficult to disentangle or even detect. We consider the scenarios presented in [26] as a reference to identify common real-world scenarios, where machine learning-based decision making is used. We compare the scenarios in Table I.

Other scenarios include insurance policy prediction [38], income prediction [28], equal opportunity policies for health care [33], teacher evaluation and promotion [9], online recommendation [23], and university ranking [27], [34].

With the rise of data science and machine learning in recent years, research interest in statistical notions of fairness has increased. Here, the most prominent examples are *group fairness* and *individual fairness* [11]:

- *Group fairness* intuitively stipulates that groups that are separated by protected properties (such as gender) are to be treated in the same manner, i.e. that outcomes must not differ, given everything else is equal between the groups.
- *Individual fairness* intuitively stipulates that individuals that are similar given their non-protected properties should be treated in a similar manner.

Recent works attempt to reconcile the supposed conflict between group and individual fairness, but also call into question the sufficiency of the statistical measures that operationalize the concepts, and in particular individual fairness. For example, claims of individual fairness can also exacerbate existing biases that may then be reflected in the selection of desirable, non-protected properties [14]. Furthermore, decisions made to mitigate bias are not value-free [1].

Still, tools for operationalizing fairness, such as IBM's *AI Fairness* 360 [4], Google's *What-if* tool [39], and Microsoft's *Fairlearn* [7], depend on these highly specific statistical formalizations that reflect group or individual fairness notions. They also assume that high-quality data is available in a rather unambiguous context that allows for the societally beneficial operationalization of fairness using these notions. Considering the recent academic discourse on the diversity and heterogeneity of fairness definitions that are needed to facilitate nuanced analysis and ultimately outcomes that are societally desirable [2], [14], it is striking that there are no formal meta-models of fairness that can instantiate a broad range of fairness definitions and scenarios from different points of view.

III. Formalization and Representation

Since our objective is to introduce an implementable and ultimately operationalizable approach to instantiate and compare context-dependent fairness definitions, our fairness formalization is grounded in conceptual approaches to fairness of societal relevance. As observed in the previous section, fairness typically pertains to decisions or actions that are made based on the attributes of specific agents or groups thereof. Each decision or action has a resource allocation or score as an outcome. Decisions or actions can be abstract, e.g., the execution of an action can be seen as assigning a score or as the use of a resource. Somewhat reflecting this intuition, we previously introduced ACROCPoLis, a conceptual framework for making sense of fairness [2].

TABLE I
COMPARISON OF REAL-WORLD SCENARIOS.

Scenario	Relevant Attributes (Input)	Outcome (Output)
Job hiring	affiliation, education level, job experience, IQ score, age, gender, address	a decision and/or a score
Granting loans	credit history, purpose of the loan, loan amount requested, employment status, income, marital status, gender, age, address, housing status, and credit score	decision and/or score
College admission	institutions previously attended, SAT scores, extracurricular activities, GPAs, test scores, interview score	decision or score
Criminal risk assessment	number of arrests, type of crime, address, employment status, marital status, income, age, housing status	score and decision
Child maltreatment prediction	contemporaneous and historical information for children and caregivers	score (likelihood) and decision
Health care	disease (chronic conditions) prediction include vital signs, blood test, sociodemographic data, education, health insurance, home ownership, age, race, address	score (likelihood)
Facial analysis	face (image)	decision

ACROCPoLis identifies six entities that are general to model fairness scenarios: *Actors*, *Context*, *Resources*, *Outcome*, *Criteria*, and *Power*, as well as the *Links* connecting them. In order to make the ACROCPoLis framework usable, we made decisions on the formalization, which required a trade-off between simplicity and generality. In our approach, we consider Actors, Context, Resources, and Outcome, and we add Measure, Aggregation, and Attribute, as we describe in Table II. We encode Criteria, Power, and Links indirectly in the other entities. Criteria are the explicit or implicit aspects needed to make a decision, affect, or justify the outcome. We interpret Power as an attribute of actors, which could be indirectly used from the Context. Links are the relations included in the attributes and in the aggregations.

This section introduces our formal meta-model of fairness and explains how the meta-model can be applied to instantiate fairness scenarios, with the notation that we provide.

A. Meta-model

Our meta-model requires two sets: I, which is a non-empty set of identifiers, and M, which is a non-empty set of measures. For the set of identifiers I, we also require a relation '$\leq$' that is a *total order*. This means that, for every $a_1, a_2, a_3 \in I$,

1) $a_1 \leq a_1$ (reflexive);
2) if $a_1 \leq a_2$ and $a_2 \leq a_3$, then $a_1 \leq a_3$ (transitive);
3) if $a_1 \leq a_2$ and $a_2 \leq a_1$, then $a_1 = a_2$ (antisymmetric);
4) $a_1 \leq a_2$ or $a_2 \leq a_1$ (strongly connected).

Some data types that could implement I are a set of strings with alphabetical order, or a set of integers with a 'less than or equals to' relation, or any other possibly infinite set with a total order.

For the set of measures M, we require it to be a subset of the real numbers $\mathbb{R}$ enriched with a distinguished element `NaN` ('Not a Number'), with the usual total order '$\leq$' for $\mathbb{R}$, and basic operations, like addition, subtraction, multiplication, and division. M could be implemented by a floating point data type [22]. In fact, `NaN` is a particular value of numeric data types, such as the floating point number, and captures cases where operations on floating point are undefined, e.g., when dividing by 0.

Once I and M are defined, we can identify a specific fairness scenario, which we call a *context*, and we just use an identifier $c \in I$ to refer to this. We do not need more structural information regarding the context, because all the relevant information of the context is in fact in other components of the tuple. Similarly to the case of the context, we identify the actors and resources by their identifiers, allowing functions on them to provide relevant information about them. The set of actors is Ac and the set of resources is R, and both are subsets of I, i.e. $Ac \subseteq I$ and $R \subseteq I$. We also require that there are no common identifiers in both sets, and that both do not contain c, i.e. $Ac \cap R = \emptyset$ and $c \notin Ac$, $c \notin R$.

Up to this point, we have defined the basic sets of identifiers (I) and measures (M), and some relevant elements of I, such as the context c, the elements of Ac and the elements of R. With these defined, we can define a set of attributes, which we call At. This set is in fact a finite set of functions f that take an identifier in Ac or R, and return either another identifier in Ac or R, or a measure in M. To denote this, we define $Fun(A, B)$ as the set of functions from A to B:

$$Fun(A, B) := \{f \mid f : A \to B\}.$$

Then, we require that the following holds:

$$\begin{aligned} At \subseteq\ & Fun(Ac, Ac) \cup Fun(R, Ac) \cup \\ & Fun(Ac, R) \cup Fun(R, R) \cup \\ & Fun(Ac, M) \cup Fun(R, M). \end{aligned}$$

We define the set of aggregation functions as a finite and possibly empty set Ag that contains only functions that can operate on any finite sequence of elements in either identifiers in Ac, identifiers in R, or measures in M, and return a single element of the same set as the domain. This can be denoted as follows. Let $Agg_n(A)$ be defined as the set of functions in sequences of elements of A of length n to an element of A, denoted by:

$$Agg_n(A) := \{f \mid f : A^n \to A\},$$

TABLE II
ENTITIES

Entity	Meaning	Relation to ACROCPoLis
Actor	is an individual or organization that participates in the fairness scenario, either by receiving resources, distributing resources, or affecting the distribution of resources.	the same as *Actor*
Context	is an entity that contains relevant contextual and structural factors in a fairness scenario.	the same as *Context*
Resource	is a measurable element to be distributed to the actors involved in a fairness scenario.	the same as *Resource*
Outcome	is the association between actors and resources in a fairness scenario.	the same as *Outcome*
Measure	is the space of quantities and qualities to measure and compare attributes of context, actors, and resources.	part of *Links*
Aggregation	is the space of functions to combine quantities and qualities and preserve them as measures.	part of *Links*
Attribute	is the space of concrete relevant features of an actor, a resource, or the context, especially reflecting a quantity or a quality.	part of *Links*, covering *Power*

where A^n denotes the n-ary Cartesian power of A. Then, we say that:

$$Ag \subseteq \bigcup_{k\in\mathbb{N}} (Agg_k(Ac) \cup Agg_k(R) \cup Agg_k(M))$$

We can define the outcome O of a scenario of fairness as a finite possibly empty set of pairs, each pair called an *assignment*, where each actor receives one resource. We can denote this as $O \subseteq \{\langle a, r\rangle \mid a \in Ac, r \in R\}$. This outcome is to be evaluated to determine whether it is fair or not according to the definition of fairness defined by human evaluators.

Given that the components are defined above and assuming that Ac, R, O, M, Ag, and At are all pairwise disjoint, we can define the tuple for a given scenario of fairness as:

$$F_c = \langle Ac, R, O, M, Ag, At\rangle.$$

We name the whole framework above AcROMAgAt. Note that I is only indirectly mentioned through its relevant elements, namely c, the elements in Ac, and the elements in R.

B. Steps to identify the entities

As described above, resource allocation scenarios are intended to allocate limited resources among actors. To identify the abstract components in this kind of scenario, we want to model whether a particular resource allocation satisfies the needs of actors according to our definition of fairness. To illustrate our definitions, we consider the entities involved in modeling a childcare subsidy scenario.

The first step is to recognize the *actors*, the *resources*, and the *context*. It might be the case that, for a given scenario, some actors are not visible or not clearly identifiable, but we focus on those receiving the resources in a particular context. In the case of the childcare subsidy scenario, each actor would be a family, the resources would be the amount paid, and the context the name of the country or territory where the subsidy is being considered.

We can then recognize the *attributes* of actors and resources that are relevant in the given context. As we learn from the requirements, some attributes would be the income of the family, the number of children, and their ages. Attributes for the resources could be the amount paid, and the currency. The *outcome* can be defined considering actors and resources, and the *measures* are those quantities and qualities that emerge from the attributes. The outcome represents how much is given to each family. Lastly, we identify *aggregations* to combine quantities and qualities and compare them. Aggregations can be seen as a collection of utility functions that help express qualities and quantities as functions of basic values. For example, if a family receives multiple childcare subsidies instead of one, an aggregation function can ensure that the total amount does not exceed the established cap per family.

In the case of scoring scenarios, the steps are analogous, but there is an emphasis on the role played by the attributes, since the score is what is being scrutinized for fairness. As in the case of resource allocation, context attributes provide the required additional information, such as historical information. At first, we could consider scoring as the allocation of an unlimited resource, but it is a limited resource in some cases, as when choosing a candidate for a job interview, or when it is used in an examination that is later normalized among all results to follow a statistical distribution. We consider scoring the allocation of an infinite abstract resource. Intuitively, there may be an overlap between scoring and resource allocation, e.g., if school grades must follow a pre-specified distribution; in our interpretation, this is *not* a scoring scenario, because the resource is finite (given a finite set of actors). The two scenario categories are not disjoint. The same problem could be modeled as a resource allocation scenario or a scoring scenario, depending on what features are more predominant or relevant for the particular use.

C. Fairness pipelines with Tiles

For modeling AcROMAgAt fairness scenarios, we introduce Tiles, which is a system to define rules based on the composition of building blocks (*tiles*). To demonstrate how Tiles work, we assume an abstract fairness scenario $F_c = \langle Ac, R, O, M, Ag, At\rangle$. Each tile has an identifier or function, an input, and an output, depicted as follows:

$$\boxed{\text{input}\ \ \text{function}\ \ \text{output}}$$

Tiles can be connected to create a *composite tile*, where the output of one tile is the input of another. They can be seen as compositions of tiles. They are connected using *connection ports* (the inputs and outputs of the function), and in some cases, a tile may have multiple input connection ports and/or multiple output connection ports. A tile with multiple input ports can be interpreted as a function with multiple parameters, or similarly of just one parameter which is a tuple of multiple ports. A tile with multiple output ports, instead, is interpreted as the replication of the output of the tile seen as a function. Multiple ports are denoted using commas, i.e. $(a_0), (a_1)$ denotes two ports of one sequence each, where both possibly empty sequences have the exact same number of elements. This allows us to re-write it as a sequence of pairs $(\langle a_0, a_1 \rangle)$.

A *pipeline* is a special case of a composite tile, which has a *starting* tile and an *ending* tile. The starting tile does not have an input, and the ending tile has a single value as output, which is usually a Boolean value. An *unfold* tile generates a sequence from a single value, for example, if given the number n, it creates a sequence of n elements. A *fold* tile generates a value from a sequence, for example, if it computes the sum of all the elements in a sequence. When configuring a pipeline, each tile can use *contextual information* and the *outcome* O all along the pipeline. The contextual information and the outcome remain constant with respect to the pipeline.

Let us see how AcROMAgAt fairness scenarios are represented by Tiles. *Actors* can be represented by the tile $\boxed{\text{all-actor}\ \ (\text{a})}$, which returns a sequence of actors, denoted by (a), i.e. $(a) = \langle a_0, ..., a_{n-1} \rangle$, where each $a_i \in Ac$, and for $1 \leq i < j \leq |Ac|$ and $a_i, a_j \in Ac$, we have $a_i \neq a_j$. This sequence is sorted by identifier.

Based on the sequence of actors, we can define a tile that retrieves the *resource* for each actor. This is achieved by the tile $\boxed{(\text{a})\ \ \text{received}\ \ (\text{m})}$, which, given an *aggregation* function $\sigma \in Ag$, $\sigma : M \to M$, and an *attribute* $p \in At$, for each a in the input sequence of actors, returns a *measure* m such that:

$$m = \sigma\ (\{p(r)\ \ |\ \ \langle a, r \rangle \in O\}).$$

To avoid verbosity in the tiles, we use the following notation conventions.

- We use a variable of a type to denote the type or the variable, depending on the context. For example, in the case of a for Ac, a can denote the type Ac or a variable of type Ac.
- We denote $(\cdot)$ as the sequence type and its elements. For example, (a) is a sequence of actors.
- We use a without index to denote an element of the sequence.
- When dealing with multiple ports, the variables in the input ports are independent from the variables in the output ports. For example, in $\boxed{(\text{m}_0),(\text{m}_1)\ \ \text{plus}\ \ (\text{m}_0)}$, the m_0 in the output port can be different from the m_0 in the input port.

The tile $\boxed{(\text{m})\ \ \text{all-equal}\ \ \text{b}}$ is true if and only if all the elements in the input sequence are equal. With the tiles defined above, we can define the tile $\boxed{\text{equality}\ \ \text{b}}$ as a pipeline as shown in Figure 1.

We can use similar definitions to encode equity, where actors receive resources according to their need, which depends on the actor and on the context, but not on the given resource.

The tile $\boxed{(\text{a})\ \ \text{needed}\ \ (\text{m})}$ is a function that, for each actor $a \in Ac$, returns the need (measure) $m \in M$ with respect to an attribute $p \in At$. The tile $\boxed{(\text{m}_0),(\text{m}_1)\ \ \text{all-at-least}\ \ \text{b}}$, given a pair of sequences, returns true if and only if for $m_0, m_1 \in M$, each pair m_0, m_1 verifies $m_0 \geq m_1$. The tile $\boxed{\text{all-actor}\ \ (\text{a}_0),(\text{a}_1)}$ works similarly to $\boxed{\text{all-actor}\ \ (\text{a})}$, but returns a pair of sequences, where each pair duplicates the same actor, for parallel processing. Figure 2 shows how we encode equity.

We see how we distinguish connections between tiles by giving subindices to their connecting variables, regardless of the fact that a_0 and a_1 are the same actor.

A tile pipeline, such as the one in Figure 2, can intuitively be seen as a directed acyclic graph, where the tiles are the vertices, the starting tiles are the source vertices, the ending tiles are the sink vertices, the edges are the connections between tiles, and the edge direction is implicit by connecting the output of one tile to the input of another.

D. Tiles *for scoring scenarios*

Based on Table I, we provide tiles centered on statistical approaches for scoring scenarios. In Figure 3, we present one possible pipeline of tiles to determine whether there is a correlation between an attribute and the performance of a prediction on individuals. Finding a correlation between values does not ensure causality, but it can serve as an indicator to detect possible unfair situations.

We assume that there is a threshold such that the values m above that threshold are positive and those below are negative. Alternatively, the implementation of these tiles could abstract such a threshold by returning Boolean values true or false. Without loss of generality, we assume that m is 0 for false and 1 for true. We use these values to calculate the Pearson correlation coefficient [37].

The tile $\boxed{\text{all-actor}\ \ (\text{a}_0),(\text{a}_1),(\text{a}_2)}$ is a tile that allows for three connection ports and produces three identical sequences of actors. The tile $\boxed{(\text{a})\ \ \text{prediction}\ \ (\text{m})}$ takes a sequence of actors, with each actor $a \in Ac$, returns the predicted values with respect to an attribute $p \in At$ as a sequence of measures $m \in M$. The tile $\boxed{(\text{a})\ \ \text{result}\ \ (\text{m})}$ takes a sequence of actors, with each actor $a \in Ac$, and returns the actual values with respect to an attribute $p \in At$ as a sequence of measures, $m \in M$. In the case of the prediction of recidivism, the *prediction* can be taken from the data two years before the evaluation and the *results* from what actually happened. Both sequences are

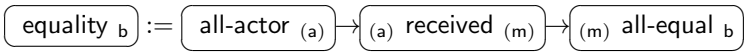

Fig. 1. Pipeline for equality: it is defined with three tiles, one producing actors, then a tile that retrieves what each actor receives, and the last one that checks whether all received the same.

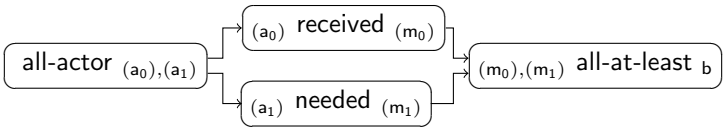

Fig. 2. Representation of equity using Tiles. The first tile on the left creates the sequence of actors that are processed in parallel, but respecting the order, by two tiles. These tiles return how much an actor received and how much the actor needs. The last tile on the right compares both values.

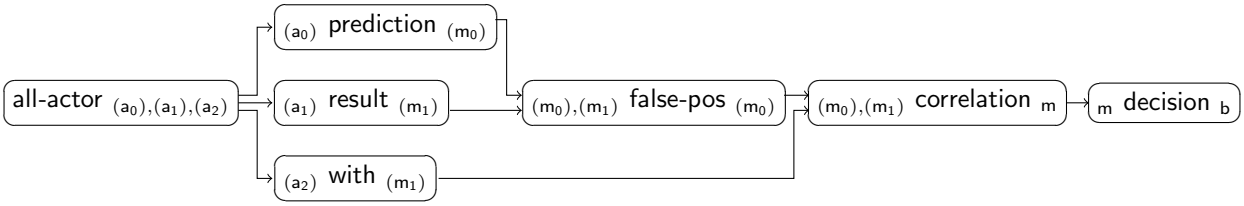

Fig. 3. Example of a configured correlation pipeline to measure the bias on false positives. The tile on the left creates triples of actors. The three branches are the original prediction on an actor ('prediction'), the actual result of an actor ('result'), and if the actor has a given property ('with'). With the original prediction and the actual result, the false positives are calculated. This, together with the characteristic of a property, is given to compute the correlation. Ultimately, we find the decision of whether there is a significant bias based on the correlation.

combined to estimate false positives, which is done by the tile (m0),(m1) false-pos (m).

The tile (m0),(m1) false-pos (m), given a pair (m_0, m_1), $m_0, m_1 \in M$, returns 1 if the pair is a false positive, and 0 otherwise. A false positive is that the prediction is 1 and the actual value is 0. The tile (m0),(m1) false-neg (m) returns 1 if the pair is a false negative, and 0 otherwise. A false negative is that the prediction is 0 and the actual value is 1. (m0),(m1) true-pos (m) and (m0),(m1) true-neg (m) are analogous, but return 1 if given (m_0, m_1), $m_0 = m_1$, and 0 otherwise. The tile (a) with (m) retrieves from all actors an attribute p, for example, the skin color. Binary attributes can be encoded with 0 and 1 to compute the correlation.

The tile (m0),(m1) correlation m computes a correlation coefficient for the subsets filtered by attributes with respect to the score. We chose the Pearson correlation coefficient, but other correlations can be used in this diagram, as long as they respect the same input/output ports. The Pearson correlation is defined, for a sample of size n, for x_i, y_i $(1 \leq i \leq n)$ individual sample points, for $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$, the sample arithmetic mean, and the same for $\bar{y}$ as follows:

$$r_{x,y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

A final tile m decision b makes the decision about whether the correlation is acceptable. For example, some arbitrary categorization could define the ranges $(0, 0.3]$ as weak correlation, $(0.3, 0.5]$ as moderate correlation, and $(0.5, 1]$ as strong correlation.

E. Implementation

Tiles can be configured for specific scenarios. Each configuration should be implemented in a more fine-grained language. Considering such a configuration, we believe that the language in which Tiles can be configured should have good readability, although this is a property that is difficult to measure. We chose SODA [32], [31] because it is an object-oriented functional language, especially designed to describe, analyze, and model human-centered problems. The tiles used in the examples are summarized in Table III, and we provide an open source implementation of them at https://julianmendez.github.io/tiles .

F. Assumptions

We assume that the information we have is *consistent*, that the resources have either a *utility function* or a *score*, and that we are provided with *complete information* of the outcome, which means that we know exactly what each actor receives. In practice, we may need to detect that a system is not fair before analyzing all assignments. Nevertheless, we can still model the problem for a particular instance at a particular point in time.

Finally, another assumption is that each tile is decidable, and that the complexity of the whole pipeline does not impede the execution possibility. Although we provide the elements to check fairness and also examples, we do not state if the

TABLE III
SUMMARY OF ACROMAGAT TILES USED IN THE EXAMPLES.

Generic Tile	**Meaning**
(α) all-satisfy p b	Given a sequence of objects of type α, it returns true if and only if all the elements satisfy property p.
$(\alpha_0),(\alpha_1)$ f(α_0,α_1) (α)	Given a pair of sequences of two objects of the same type α, it returns a sequence of objects of the same type, resulting from applying the function f to both elements of the pair. If the parameters are omitted, the order is as expected. For example, for measures, $(m_0),(m_1)$ plus (m) denotes that each element m in the output sequence is computed by applying the function plus (+) to two measures, i.e. $m = m_0 + m_1$.
(α) p? (α)	Given a sequence of objects of type α, it returns a possibly empty sequence of objects of the same type such that all of them satisfy the property p.
all-actor (a)	Returns a sorted sequence of actors (a), where each $a \in Ac$ occurs exactly once.
(a) received (m)	Given a sequence of actors (a), with $a \in Ac$, it returns a sequence of measures (m), $m \in M$, such that each m is the aggregated value using the aggregation function σ applied to the set produced by the resource attribute p, based on the outcome O.
(m) all-equal b	Given a sequence of measures (m), $m \in M$, it returns true if all values are equal.
Customized Tile	**Meaning**
(a) needed (m)	Given a sequence (a), for each $a \in Ac$, and the attribute $p \in At$, it returns a sequence of measures (m), where each $m \in M$ has the need of that actor with respect to p.
$(m_0),(m_1)$ all-at-least b	Given a pair of sequences $(m_0),(m_1)$, where each $m_0, m_1 \in M$, it returns true if for all pairs, $m_0 \geq m_1$, and it returns false otherwise.
(a) prediction (m)	Given a sequence of actors (a), it returns a sequence of measures (m), such that for each actor $a \in Ac$, for a measure $m \in M$, it holds that $m = 1$ if based on the outcome O the prediction with respect to an attribute $p \in At$ is positive, and $m = 0$ if it is negative.
(a) result (m)	Given a sequence of actors (a), it returns a sequence of measures (m), such that for each actor $a \in Ac$, for a measure $m \in M$, it holds that $m = 1$ if based on contextual information in c, the result with respect to an attribute $p \in At$ was positive, and $m = 0$ if it was negative.
$(m_0),(m_1)$ false-pos (m)	Given a pair of sequences $(m_0),(m_1)$, where each $m_0, m_1 \in M$, it returns a sequence of measures (m), $m \in M$, such that $m = 1$ if the value of $m_0 = 1$ and $m_1 = 0$, and $m = 0$ otherwise.
(a) with (m)	Given a sequence of actors (a), $a \in Ac$, it returns a sequence of measures $m \in M$ containing the characteristic value: 1 for those actors that have the attribute p and 0 otherwise.
$(m_0),(m_1)$ correlation m	Given a pair of sequences of measures, $(m_0),(m_1)$, where each $m_0, m_1 \in M$, it returns a single value $m \in M$, which is the Pearson correlation coefficient.
m decision b	Given a correlation measure $m \in M$, it returns true if and only if the correlation is considered significant.

elements we provide can model all possible fairness definitions or if it is feasible to model all possible fairness definitions.

IV. EXAMPLE

Let us consider an example to which the Tiles framework can be applied. For that, we go back to the childcare subsidy scenario. For the purpose of this scenario, a family has one or more parents or (legal) guardians, who are responsible for one or more children. Guardians may receive different childcare subsidies depending on the definition of fairness used. Some possible criteria for the amount of money that each family could receive are listed here:

- (no subsidy) no subsidy is given to any family (Figure 4);
- (per child) give to all families the same amount for each child (Figure 5);
- (per family) give the same amount of money to each family, regardless of the number of children (Figure 6);
- (single guardian) give the subsidy when the family has only one guardian (Figure 7).

In our diagrams, each actor is a family (as defined in this scenario). Some of the properties of a family are:

- number of adults: a positive integer (1 or more);
- number of children: a positive integer (1 or more);
- a (yearly) income: a non-negative integer (0 or more).

These properties are considered contextual information and do not change across the pipeline. The resource is money for the childcare subsidy, and it is represented by a non-negative integer. The measures are then non-negative integers.

V. CONCLUSION

In this paper, we have presented a formal meta-model for instantiating definitions of fairness, supported by a visualization approach and a proof-of-concept implementation. We envision the presented work as a step towards making differences

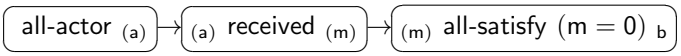

Fig. 4. Pipeline for no subsidy. The tile on the left provides all actors. The tile in the middle computes how much resource each actor received. The tile on the right checks that all resources are equal to 0.

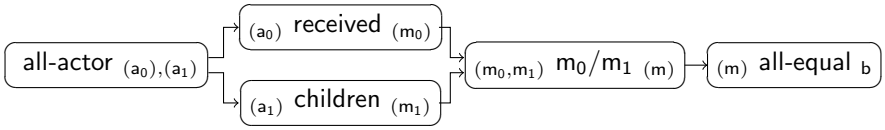

Fig. 5. Representation of "per child" using Tiles. The tile on the left provides actors, which are divided in two branches. The upper branch computes how much each actor (a family) has received and the lower branch how many children the family has. Both values are zipped back to compute the division. Note that we assume that each family has at least a child, but otherwise, if the number of children is 0, the division would be computed as `NaN`.

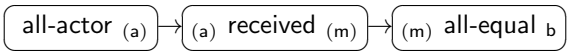

Fig. 6. Representation of "per family" using Tiles. This is equivalent to a standard equality pipeline where each actor receive exactly the same amount of resource.

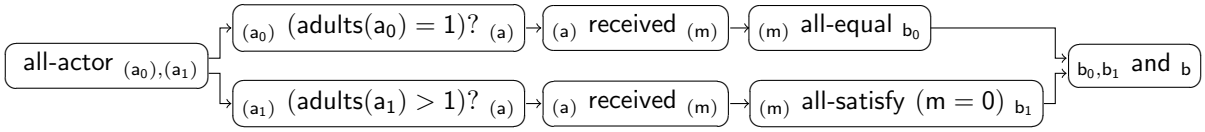

Fig. 7. Representation of "single guardian" using Tiles. This pipeline has two main branches. The upper branch accepts only families with one adult, i.e. single-parent/guardian families. The lower branch accepts all remaining families. It is worth noting that the sequences in both branches may have different number of elements and cannot be zipped back. On the other hand, the Boolean computation is combined with the 'and' tile, on the right.

between approaches to fairness in a given context explicit and qualitatively comparable.

For the next steps, our aim is to validate the framework and to expose it to domain experts and decision-makers that work on fairness-related specifications, for example, in the context of organizational and public policies, in order to elicit guidelines for practical use.

Future research can extend our work primarily in two directions. One direction from a formal perspective is to define axioms/principles for fairness scenarios. These may be related to the expected behavior of the underlying functions. For example, in a resource allocation scenario, an outcome function should exactly allocate the initially specified resources without "creating" or "wasting" any resources. Beyond that, one may specify principles that constrain subjective aspects of fairness scenarios, for instance, to gauge whether different formalizations of the same real-world scenario agree on a shared set of fundamental ideas. From an applied perspective, we aim to further advance our toolkit to define, visualize, and compare fairness definitions so that it is more accessible to practitioners such as analysts working on policy and process design, or decision automation, for example, by developing a visual interface to connect the tiles and automatically generate the source code.

Acknowledgements: This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

[1] Aler Tubella, A., Barsotti, F., Koçer, R.G., Mendez, J.A.: Ethical implications of fairness interventions: what might be hidden behind engineering choices? Ethics and Information Technology **24**(1), 12 (Feb 2022). https://doi.org/10.1007/s10676-022-09636-z, https://doi.org/10.1007/s10676-022-09636-z

[2] Aler Tubella, A., Coelho Mollo, D., Dahlgren Lindström, A., Devinney, H., Dignum, V., Ericson, P., Jonsson, A., Kampik, T., Lenaerts, T., Mendez, J.A., Nieves, J.C.: ACROCPoLis: A Descriptive Framework for Making Sense of Fairness. In: Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency. p. 1014–1025. FAccT '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3593013.3594059, https://doi.org/10.1145/3593013.3594059

[3] Barocas, S., Selbst, A.D.: Big Data's Disparate Impact. California Law Review **104**(3), 671–732 (2016). https://doi.org/10.15779/Z38BG31, http://www.jstor.org/stable/24758720

[4] Bellamy, R.K.E., Dey, K., Hind, M., Hoffman, S.C., Houde, S., Kannan, K., Lohia, P., Martino, J., Mehta, S., Mojsilovic, A., Nagar, S., Ramamurthy, K.N., Richards, J.T., Saha, D., Sattigeri, P., Singh, M., Varshney, K.R., Zhang, Y.: AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. IBM J. Res. Dev. **63**(4/5), 4:1–4:15 (2019). https://doi.org/10.1147/JRD.2019.2942287, https://doi.org/10.1147/JRD.2019.2942287

[5] Berk, R., Heidari, H., Jabbari, S., Kearns, M., Roth, A.: Fairness in Criminal Justice Risk Assessments: The State of the Art (2017). https://doi.org/10.48550/ARXIV.1703.09207, https://arxiv.org/abs/1703.09207

[6] Bin-Obaid, H.S., Trafalis, T.B.: Fairness in Resource Allocation: Foundation and Applications. In: Bychkov, I., Kalyagin, V.A., Pardalos, P.M., Prokopyev, O. (eds.) Network Algorithms, Data Mining, and

Applications. pp. 3–18. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-37157-9_1
[7] Bird, S., Dudík, M., Edgar, R., Horn, B., Lutz, R., Milan, V., Sameki, M., Wallach, H., Walker, K.: Fairlearn: A toolkit for assessing and improving fairness in AI. Microsoft, Tech. Rep. MSR-TR-2020-32 (2020)
[8] Busemeyer, M.R., Goerres, A.: Policy feedback in the local context: analysing fairness perceptions of public childcare fees in a german town. Journal of Public Policy **40**(3), 513–533 (2020). https://doi.org/10.1017/S0143814X18000491
[9] Chalfin, A., Danieli, O., Hillis, A., Jelveh, Z., Luca, M., Ludwig, J., Mullainathan, S.: Productivity and Selection of Human Capital with Machine Learning. American Economic Review **106**(5), 124–27 (May 2016). https://doi.org/10.1257/aer.p20161029, https://www.aeaweb.org/articles?id=10.1257/aer.p20161029
[10] Dator, J., Pratt, D., Seo, Y.: What Is Fairness?, pp. 19–34. University of Hawai'i Press (2006). https://doi.org/10.2307/j.ctv3zp081.6, http://www.jstor.org/stable/j.ctv3zp081.6
[11] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., Zemel, R.S.: Fairness through awareness. In: Goldwasser, S. (ed.) Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012. pp. 214–226. ACM (2012). https://doi.org/10.1145/2090236.2090255, https://doi.org/10.1145/2090236.2090255
[12] Australian Institute of Family Studies, A.G.: Understanding the Child Care Subsidy (2024), https://aifs.gov.au/research/research-snapshots/understanding-child-care-subsidy
[13] Finkel, N.J., Harré, R., Rodriguez Lopez, J.L.: Commonsense Morality Across Cultures: Notions of Fairness, Justice, Honor and Equity. Discourse Studies **3**(1), 5–27 (2001). https://doi.org/10.1177/1461445601003001001, https://doi.org/10.1177/1461445601003001001
[14] Fleisher, W.: What's Fair about Individual Fairness? In: Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society. p. 480–490. AIES '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3461702.3462621, https://doi.org/10.1145/3461702.3462621
[15] Franklin, J.S., Bhanot, K., Ghalwash, M., Bennett, K.P., McCusker, J., McGuinness, D.L.: An Ontology for Fairness Metrics. In: AIES 2022 - Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society. pp. 265–275. Association for Computing Machinery, Inc (7 2022). https://doi.org/10.1145/3514094.3534137
[16] Friedman, B., Kahn, P., Borning, A.: Value Sensitive Design: Theory and Methods. University of Washington technical report **2**, 12 (2002), https://dada.cs.washington.edu/research/tr/2002/12/UW-CSE-02-12-01.pdf
[17] Gastwirth, J.L.: A General Definition of the Lorenz Curve. Econometrica **39**(6), 1037–1039 (1971). https://doi.org/10.2307/1909675, http://www.jstor.org/stable/1909675
[18] Gastwirth, J.L.: The Estimation of the Lorenz Curve and Gini Index. The Review of Economics and Statistics **54**(3), 306–316 (1972). https://doi.org/10.2307/1937992, http://www.jstor.org/stable/1937992
[19] Gini, C.: Sulla misura della concentrazione e della variabilita dei caratteri. Atti del Reale Istituto Veneto di Scienze, Lettere ed Arti **73**, 1203–1248 (1914), https://cir.nii.ac.jp/crid/1573105974129324928
[20] Gini, C.: Measurement of Inequality of Incomes. The Economic Journal **31**(121), 124–125 (03 1921). https://doi.org/10.2307/2223319, https://doi.org/10.2307/2223319
[21] Government, M.: Child Care Subsidy (2024), https://www.gov.mb.ca/education/childcare/families/childcare_subsidies.html
[22] IEEE: IEEE Standard for Floating-Point Arithmetic. IEEE Std 754-2008 pp. 1–70 (2008). https://doi.org/10.1109/IEEESTD.2008.4610935
[23] Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: Recommender systems: an introduction. Cambridge University Press (2010)
[24] Katoh, N., Ibaraki, T.: Resource Allocation Problems, pp. 905–1006. Springer US, Boston, MA (1998). https://doi.org/10.1007/978-1-4613-0303-9_14, https://doi.org/10.1007/978-1-4613-0303-9_14
[25] Kusner, M.J., Loftus, J., Russell, C., Silva, R.: Counterfactual fairness. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30, pp. 4067–4077. Curran Associates, Inc. (2017), https://proceedings.neurips.cc/paper/2017/file/a486cd07e4ac3d270571622f4f316ec5-Paper.pdf
[26] Makhlouf, K., Zhioua, S., Palamidessi, C.: On the Applicability of Machine Learning Fairness Notions. SIGKDD Explor. Newsl. **23**(1), 14–23 (may 2021). https://doi.org/10.1145/3468507.3468511, https://doi.org/10.1145/3468507.3468511
[27] Marope, P.T.M., Wells, P.J., Hazelkorn, E., et al.: Rankings and accountability in higher education: Uses and misuses. Unesco (2013)
[28] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., Galstyan, A.: A Survey on Bias and Fairness in Machine Learning (2019). https://doi.org/10.48550/ARXIV.1908.09635, https://arxiv.org/abs/1908.09635
[29] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., Galstyan, A.: A Survey on Bias and Fairness in Machine Learning. ACM Comput. Surv. **54**(6) (jul 2021). https://doi.org/10.1145/3457607, https://doi.org/10.1145/3457607
[30] Mehrotra, A., Celis, L.E.: Mitigating Bias in Set Selection with Noisy Protected Attributes. In: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. p. 237–248. FAccT '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3442188.3445887, https://doi.org/10.1145/3442188.3445887
[31] Mendez, J.A.: Soda (2020), https://julianmendez.github.io/soda
[32] Mendez, J.A.: Soda: An Object-Oriented Functional Language for Specifying Human-Centered Problems (2023). https://doi.org/10.48550/arXiv.2310.01961
[33] Moreno-Ternero, J.D.: On the design of equal-opportunity policies. Investigaciones económicas **31**(3), 351–374 (2007)
[34] O'Neil, C.: Weapons of math destruction. How Big Data Increases Inequality and Threatens Democracy. Crown (2016)
[35] Pigmans, K., Dignum, V., Doorn, N.: Group proximity and mutual understanding: measuring onsite impact of a citizens' summit. Journal of Public Policy **41**(2), 228–250 (2021)
[36] de Reuver, M., van Wynsberghe, A., Janssen, M., van de Poel, I.: Digital platforms and responsible innovation: expanding value sensitive design to overcome ontological uncertainty. Ethics and Information Technology **22**, 257–267 (2020)
[37] Rodgers, J.L., Nicewander, W.A.: Thirteen Ways to Look at the Correlation Coefficient. The American Statistician **42**(1), 59–66 (1988). https://doi.org/https://doi.org/10.2307/2685263, http://www.jstor.org/stable/2685263
[38] Shrestha, Y.R., Yang, Y.: Fairness in Algorithmic Decision-Making: Applications in Multi-Winner Voting, Machine Learning, and Recommender Systems. Algorithms **12**(9) (2019). https://doi.org/10.3390/a12090199, https://www.mdpi.com/1999-4893/12/9/199
[39] Wexler, J., Pushkarna, M., Bolukbasi, T., Wattenberg, M., Viégas, F.B., Wilson, J.: The What-If Tool: Interactive Probing of Machine Learning Models. IEEE Trans. Vis. Comput. Graph. **26**(1), 56–65 (2020). https://doi.org/10.1109/TVCG.2019.2934619, https://doi.org/10.1109/TVCG.2019.2934619