PROPERTY THEORY: AN INTRODUCTION

VIPUL NAIK

Abstract. "Property theory", as I call it, refers to a theme – the centrality of properties, and to the tools and formalisms that I have evolved around that theme. These tools, that I originally developed to get a better understanding of some areas of mathematics, have taken a life of their own. In this article, I describe my development of the theory as I have developed. The purpose is both to explain the theory and to collect feedback about its importance, understand its relation with existing frameworks and get ideas on how to develop it further.

1. What is property theory?

1.1. **Understanding through properties.** The central theme of property theory is – *treat properties as elements in their own right*. Manipulate the properties themselves, rather than the elements having those properties. Ask questions about the properties. Start with properties to get new properties.

One of the new paradigms in programming has been the so called *functional programming paradigm*. What is the essence of functional programming? It permits us to treat functions as *first class objects*, that is, it views *functions as elements in their own right*. A whole new way of thinking rests on this basic premise.

Property theory rests on an even more tempting premise – viewing properties as elements in their own right. Properties are even more fundamental than functions, and are found not only in mathematics, but cutting across all disciplines.

When I first started developing the formalisms, I did not think of stating the centrality of properties as the underlying theme. However, with the passage of time, I realized that the primary obstacle that I had in explaining the theory was that I could not put across the idea that the operations and phenomena that I was describing were being studied, not over the objects themselves, but over their properties. This led me to realize that what was really important in my theory was that properties formed the vantage point.

1.2. **Basic definition of a property.** The word *property* has a variety of meanings in the English language. In the mathematical sense, however, the term has a relatively clear meaning.

Definition. A **concrete property**(defined) over a collection of objects is a map from that collection to the two element set comprising `true` and `false`. Elements mapped to `true` are said to *satisfy* the property and elements mapped to `false` are said to *not satisfy* the property.

The collection over which this map is defined is termed the **context space**(defined) of the property. The collection of *all* properties over a given context space is termed the **complete property space**(defined) over the context space.

A **concrete property space**(defined) is a subcollection of the complete property space over some context space, that contains the property satisfied by all elements (often called the **tautology**(defined)) and the property satisfied by no element (often called the **fallacy**(defined)).

Now, the first question the above definitions raise is: what is really gained by talking of properties? Because each property simply classifies the original collection into two parts, would it not be better to simply identify the property with the subcollection satisfying it? From that viewpoint, the complete property space over a context space is simply the collection of all its subcollections (or, an analogue of the **power set**).

While mathematically it seems equivalent, this view does not capture the *spirit* of properties – which is that they are ways of *separating* and *distinguishing* objects. Every property distinguishes the *have*s (objects that have the property) from the *have not*s (objects that do not have the property).

Upshot. The subcollection view of the property is definitely a starting point, but a lot of perspective is added when we introduce additional structure on the space.

1.3. **Features of properties.** Concrete properties, as defined in the previous subsection, are:

(1) **Defined over a fixed context space**
(2) **Binary**: For every element over the context space, it can take only two possible values – `true`, and `false`.
(3) **Definite**: Every object evaluates to either `true` or `false`.
(4) **Intrinsic**: The property depends on the inherent nature of the object and not on any other factors.

The notion of intrinsicity attains significance when the context space comes equipped with notions of isomorphism or equivalence. Here are some examples:

- When mathematicians study properties over the context space of the objects of a category, they would like the properties to be *preserved under isomorphisms.*
- When mathematicians study properties over the context space of the morphisms of a category, they would like the properties to be *preserved under commuting isomorphisms.*
- When logicists want to study the properties of logical structures, they would like the properties to be *preserved under isomorphisms.*
- If they are interested only in studying first order phenomena, then they may like the properties to be preserved under *elementary local isomorphisms.*
- A computer scientist may want to study properties of a function (in a programming language) that are *preserved upon α equivalences*, while another might like to study properties that are preserved upto η equivalences.

Properties preserved under certain isomorphisms or equivalences, can essentially be thought of as properties on the quotient of the original collection by those isomorphisms or equivalences. Thus, properties over a category's objects are better thought of as defined over isomorphism classes of objects, properties at the first order level over logical structures can be thought of as properties of elementary equivalence classes of structures.

1.3.1. *Properties and relations.* Relations are the multi dimensional analogues of properties. In fact, a relation over two (or more) collections can be viewed as a property over their Cartesian product. Just like properties, a relation can also be identified with the subcollection that satisfies it.

However, as we had remarked from properties, the role of a relation is as a *separating* or *distinguishing* criterion – it separates the *have*s from the *have not*s. The *have*s in this case are the tuples that satisfy the relation, while the *have not*s are the tuples that do not.

1.4. **The property space.** The previous subsection views properties as functions from a collection to the set comprising `true` and `false` This is a *concrete* picture of a property space. However, for various reasons, we may want to study the property space without bothering about its concrete manifestations – that is, we are interested in an *abstract* view of the property space. The utility of this approach shall become clearer with time. Here is the definition:

Definition. An **abstract property space**(defined) is a partially ordered collection of elements with an upper bound, called the **fallacy**(defined), denoted by f, and a lower bound, called the **tautology**(defined), denoted by t. The partial order is denoted as $\leq$ or $\implies$.

Earlier on, we had talked of concrete property spaces that are subcollections of the complete property space over a context space. Every concrete property space is naturally an abstract property space. The partial order $\leq$ is the partial order of inclusion or implication. We say that $a \leq b$ or $a \implies b$ (Where a and b are properties in a concrete property space) if any of these equivalent conditions hold:

- Every element in the context space satisfying a satisfies b
- The property of satisfying a is sufficient for satisfying b, that is, it is *stronger*
- The property of satisfying b is necessary for satisfying a, that is, it is *weaker*
- The subcollection satisfying a is *contained in* the subcollection satisfying b. Thus, it is in some sense *smaller*, explaining the $\leq$ sign
- The condition of an element in the context space satisfying a implies the condition of satisfying b, explaining the $\implies$ sign

If we were to identify concrete properties with the subcollections satisfying them, then the $\leq$ would simply be the containment partial order.

But this way of identifying properties relates a property to the proposition that a given (unspecified) element of the context space satisfies the property. Thus, *logical operations* performed on propositions can now be done on properties. Thus, we can borrow from logical theories to study the property space.

Upshot. There are two approaches to concrete properties:

- **As subcollections**: Here, the concrete property is identified with the subcollection of the property space that satisfies it.
- **As logical propositions with a parameter**: Here, the concrete property is identified with the logical proposition stating that a given element of the context space satisfies the property. That "given element" is the parameter for the logical proposition. Thus, the logical proposition corresponding to property p is the statement $x \models p$ where x is the parameter from the context space.

The choice of terminology and notation for abstract properties is reflective of both these approaches for their concrete counterpart.

1.5. **What is really new?** The *really new* thing about property theory begins with our trying to put operators and algebraic structures on the property space. The idea: define n ary operations from a property space to itself, define operations from one property space to another, and so on. These operations can be viewed in terms of the perspective that properties are subcollections, and they can also be viewed in terms of the perspective that a property is identified with a proposition of its being satisfied.

However, the richness of formalism and the refreshing similarity between the many property spaces that arise in nature suggests that the logical proposition viewpoint, while a good starter, is not the full story so far as property theory is concerned. In fact, as I have gradually unfolded the ideas of property theory, I have found myself acquiring a new and different kind of intuition.

1.6. **Some examples of property spaces.** Each of examples of property spaces that I'll talk about now will be developed in more detail as we proceed to acquire more and more property theoretic tools. We include a preliminary discussion of these.

1.6.1. *Groups.* "Groups" form a category (for the category theorist). They also have a first order theory (for the logicist and model theorist). In fact, they are a "variety of universal algebras" (for the algebraist).

The **group property space**(defined) is the complete property space over the context space of isomorphism classes of groups. Note that the properties are defined over *isomorphism classes* of groups, that is, we can think of them as *isomorphism invariant* properties of groups.

The isomorphism classes of groups do not form a **set**(first used), rather, they form a so called **class**(first used), which is a collection only slightly bigger than a set. So, the property space over this is bigger than any set ever could be. That is, it's really huge. Yet, will we ever even begin to touch all those properties? Can we *describe* all of them? No.

This suggests that we can restrict ourselves to a smaller property space – the space of **definable**(defined) properties over the context space of groups. These are properties for which we can give a finite description using the language of set theory.

The good thing is that the definable property space will be closed under all definable operations, and also, that it is countable. So, we might as well work in the definable property space for most purposes. However, there are a few caveats, that I'll come to later.

Here are some group properties (and hence, elements of the group property space):

- The property of being *Abelian*
- The property of being *cyclic*
- The property of being *solvable*
- The property of being *perfect*
- The property of being *finite*
- Th property of being *trivial*

Note that the property of being *normal* is not a group property, because given a group it does not make sense to ask whether or not it is normal. Rather, it is a property that depends on the datum of a subgroup inside another group. Hence, normality lies in the subgroup property space – which forms the content of the next subsubsection.

1.6.2. *Subgroups.* Here, we are considering properties that, given a group and a subgroup, are either true or false for the pair.

The tricky point here is the notion of *isomorphism* to make the properties *intrinsic*. When is a pair $H_1 \leq G_1$ same as a pair $H_2 \leq G_2$? Clearly, $H_1 \cong H_2$ and $G_1 \cong G_2$, but these conditions are not sufficient to ensure that the behaviour of H_1 inside G_1 is the same as that of H_2 inside G_2. What we need is an isomorphism of G_1 with G_2 and of H_1 with H_2 such that the isomorphisms *commute* with the inclusion maps. We shall call such a pair of isomorphisms *commuting isomorphisms*.

The **subgroup property space**(defined) is the complete property space over the context space of group-subgroup pairs, modulo the equivalence relation of commuting isomorphisms. In other words, the subgroup property space is the space of properties over group-subgroup pairs that are preserved upon performing commuting isomorphisms.

Here are some subgroup properties (and hence, elements in the subgroup property space):

- The property of being *normal*
- The property of being the *whole group*, or of being the *improper* subgroup
- The property of being *characteristic*
- The property of being a *direct factor*
- The property of being *fully invariant*
- The property of being a *retract*
- The property of being a *central factor*
- The property of being a *Sylow subgroup*
- The property of being the *trivial subgroup*

1.6.3. *Graphs.* The **simple graph property space**(defined) is the complete property space over the context space of isomorphism classes of simple graphs. By *simple graph*, I mean a pair (V, E) where V is a set and E is a set whose elements are unordered pairs of distinct elements of V.

As in the case of groups, the simple graph property space gives those properties of simple graphs that are *isomorphism invariant*. Here, an isomorphism of two graphs is a bijection of their vertex sets and their edge sets such that the image of the edge joining two vertices is the edge joining their images.

Here are some properties of simple graphs (and hence, elements of the simple graph property space). Note that each of them is binary, definite and intrinsic:

- The property of being *cyclic*
- The property of being *regular*
- The property of being *connected*
- The property of being *triangulable*
- The property of being *finite*
- The property of being *locally finite*

Note that *directedness* is not a property of simple graphs – rather, introducing directions is a structural **augmentation**(first used) of simple graphs – we are adding new data to the original structure. However, *orientability* is a valid property because here, what we claim is that *there exists* an orientation satisfying certain conditions.

1.6.4. *Formal languages.* A formal language is a subset of the set of all strings over a finite alphabet. Two formal languages are termed isomorphic if there is a bijection between the set of letters used in each that induces a bijection between the formal languages themselves.

The **formal language property space**(defined) is the complete property space over the context space of isomorphism classes of formal languages. That is, it is the space of *isomorphism invariant* properties of formal languages.

Here are some properties of formal languages, that are hence elements of the formal language property space:

- The property of being *finite*
- The property of being *star free*
- The property of being *non empty*
- The property of being *the whole language* on the alphabet of used letters
- The property of being *context free*
- The property of being *unambiguously context free*
- The property of being *recursively enumerable*

1.6.5. *Topological spaces.* A **topological space**(defined) is a set (the *space*) along with a collection of open subsets that contain the empty set and the whole space, are closed under finite intersections, and are closed under arbitrary unions. The notion of isomorphism for topological spaces is called **homeomorphism**(defined) – a bijective map that is continuous both ways, that is, a bijection of the underlying sets where the open sets in one correspond precisely with the open sets in the other.

The **topological space property space**(defined) is the complete property space over the context space of homeomorphism classes of topological spaces. That is, it is the space of *homeomorphism invariant* properties of topological spaces.[1]

Here are some topological space properties (and hence, elements of the topological space property space):

- The property of being *Hausdorff*
- The property of being *regular*
- The property of being *compact*
- The property of being *connected*
- The property of being a *Baire space*
- The property of being *normal*
- The property of being *metrizable*
- The property of being *paracompact*
- The property of being *linearly orderable*

Note that the property of being open or closed is *not* a property of topological spaces – rather it is a property of *subsets* of topological spaces.

1.6.6. *Other examples.* Here are some other examples of property spaces that we shall not carry along in great detail, but shall refer to from time to time:

- The context space being *complexes of Abelian groups*, with the notion of equivalence being *quasi isomorphism.*
- The context space being *coverings of a topological space*, with the notion of equivalence being homeomorphism of the topological space that maps each covering set to a covering set.
- The context space being *models of first order theories*, with the notion of equivalence being elementary equivalence.

1.7. Metaproperties.

Metaproperties are *properties of properties.* They are concrete properties whose context space itself is a property space.

Definition. A **metaproperty**(defined) is a concrete property whose context space is itself a property space. When we talk of a metaproperty over a context space, we mean a property over the complete property space over that context space.

A little care about terminology. A **concrete metaproperty**(defined) is a concrete property whose context space is a *concrete* property space, and the context space of this property space is what we call the "context space of the metaproperty". Thus, its context space *as a metaproperty* differs from its context space as a property – the context space as a metaproperty goes two levels down, while the context space as a property goes one level down.

Given a metaproperty over a property space, we get a smaller property space – the space of properties that satisfy the metaproperty. If the original property space was the complete property space over some context space, we get a concrete property space. The examples shall make this idea clear.

As we shall see, the complete property space, or even the space of definable properties, is too huge for our liking, and one of the goals of developing property theoretic tools and formalisms is to find metaproperties that filter out the *nice* and *well behaved* properties from the remaining ones. The notion of niceness may differ based on the kind of application we seek. A mathematician may be interested in *set theoretic definability*, while a program designer might be interested in *computability using reasonable space and time resources.* Some people may be interested in certain *closure properties* or *regularity properties* that make proofs easier.

Let's look at the examples introduced in the previous section and introduce metaproperties over them.

[1]Algebraic topologists often consider this property space too huge. They prefer to work with *homotopy invariant* properties of topological space. Homotopy invariance restricts the nature of topological properties that can be studied.

1.7.1. *Groups.* Here are some metaproperties over groups. They can be viewed as properties whose context space is the group property space.

- The metaproperty of being *hereditary.* A group property is said to be **hereditary**(defined) if every subgroup of a group having the property also has the property.
- The metaproperty of being *quotient hereditary.* A group property is said to be **quotient hereditary**(defined) if every quotient group of a group having the property also has the property.
- The metaproperty of being *preserved under extensions.* A group property is said to be **extension preserved**(defined) if the extension of a group having the property by another group having the property also has the property.
- The metaproperty of being *varietal.* A group property is said to be **varietal**(defined) if it is preserved under taking subgroups, quotients, and arbitrary direct products.
- The metaproperty of being *asymptotic.* A group property is said to be **asymptotic**(defined) if any subgroup of finite index in a group with the property, also has the property.

Each of these metaproperties can give rise to a subspace of the group property space – namely, the subspace comprising those properties that satisfy the given metaproperty. For instance, the metaproperty of being hereditary gives rise to the concrete property space comprising hereditary group properties.

1.7.2. *Subgroups.* Here are some subgroup metaproperties. The corresponding concrete property spaces shall attain added significance as we proceed towards property operators.

- The metaproperty of being *transitive.* A **transitive subgroup property**(defined) is a subgroup property such that if H has the property as a subgroup of K and K has the property as a subgroup of G, H has the property as a subgroup of G.
- The metaproperty of being *identity true.* A subgroup property is said to be **identity true**(defined) if every group satisfies the property when viewed as a subgroup of itself.
- The metaproperty of being *lower meet preserved.* A subgroup property is said to be **lower meet preserved**(defined) if the meet (intersection) of two subgroups having the property in a group also has the property in the group.

1.7.3. *Graphs.* Some interesting graph metaproperties are:

- The metaproperty of being *subgraph hereditary.* A **subgraph hereditary graph property**(defined) is a property such that any induced subgraph of a graph with the property also has the property.
- The metaproperty of being *monotone.* A **monotone graph property**(defined) is a graph property such that the addition of edges to a graph with the property gives another graph with the property.
- The metaproperty of being *true for almost all graphs.* A graph property is said to be true for almost all graphs if the limiting value of the ratio of number of graphs with n vertices satisfying the property, to the total number of graphs with n vertices, is 1.

1.7.4. *Formal languages.* Some interesting formal language metaproperties:

- The metaproperty of being *grammar describable.* A grammar description of a formal language property is a general grammar format such that any language with the property has a generating grammar satisfying that format. A formal language property is said to be **grammar describable**(defined) if it has a grammar description.
- The metaproperty of being *preserved under homomorphic images.* If the image of any formal language with the property also has the property, it is considered as being **preserved under homomorphic images**(defined).
- The metaproperty of being *closed under concatenation.* A formal language is said to be closed under concatenation if the concatenation of any two words in the language is also in the language.

1.7.5. *Topological spaces.* Important topological space metaproperties:

- The metaproperty of being *weakly hereditary.* A topological space property is said to be **weakly hereditary**(defined) if every closed subspace of a space with the property also has the property.
- The metaproperty of being *preserved under products.* A topological space property is said to be **preserved under products**(defined) if the product of topological spaces, each with the property, also has the property, when viewed in the product topology.

1.8. **A summary so far.** I have defined a **concrete property**(recalled) as a map from a collection (called the **context space**(recalled)) to a two element set comprising `true` and `false`. Properties over a context space can be identified with subcollections of the space by identifying a property with the subcollection that is mapped to `true`. Every property can also be identified with a parameterized logical proposition.

Rather than studying each property in isolation, we would like to study the space of all properties over a given context space. We called this the **complete property space**(recalled). However, *all* properties could get too messy, so we might like to restrict to only those properties that satisfy certain nice **metaproperties**(recalled). A property space obtained as a subspace of a complete property space is termed a **concrete property space**(recalled).

What we'll find, as we proceed along, is that the view of the complete property space (or a nice concrete property space), on the *whole*, is a far more powerful aid to intuition than a view of each property space in isolation, or the view of properties simply as subcollections. The primary reason for this is that we can define *algebraic* operations on the structure, and formulate known results in terms of *equations* and *inequations* in the operations we define.

1.9. **Metametaproperties: an addendum.** The abstraction from properties to metaproperties can be carried one level further to metametaproperties. I haven't used these explicitly (in fact, this article is the first time I am discussing them) but there are certain situations where they do pop up.

A simple way of explaining this is by recalling that:

> Every metaproperty gives rise to a subspace of the property space by restricting to those properties that satisfy the metaproperty.

We often use metaproperties to help us restrict the property space meaningfully in such a way that we get to deal with a *nicer* and *smaller* collection of properties. To do this, we need to ensure that the metaproperties themselves satisfy some nice properties. This is where metametaproperties come in: in regulating the choice of metaproperties that we use.

2. Logical operations in the property space

2.1. **Conjunction.** The **meet**(defined) of two elements in a partially ordered set is the largest element that is bounded above by both of them. A poset where any two elements have a well defined meet is termed a **meet semilattice**(defined). The meet is denoted by the symbol $\wedge$. Thus, $a \wedge b$ is the property c such that $c \leq a$, $c \leq b$ and if $d \leq a$, $d \leq b$, then $d \leq c$.

When a property space is a meet semilattice, the meet operation is called **conjunction**(defined), and the property space is termed a **conjunctive property space**(defined).

For a complete property space, the conjunction operation can be understood by either of the two interpretations for properties seen so far:

- **As subcollections**: The conjunction of two properties corresponds to the subcollection that is the intersection of the subcollections corresponding to the two properties. This follows from two facts:
  - If we set c to be the property of being in the intersection of the subcollection satisfying a and the subcollection satisfying b, then $c \leq a$ and $c \leq b$.
  - If $d \leq a$, and $d \leq b$, then every element satisfying d must lie in the intersection of the subcollections satisfying a and b. Hence $d \leq c$.
- **As logical propositions**: The conjunction of two properties corresponds to the *logical conjunction*, or the And, of the corresponding propositions. This follows from two facts:
  - if we set c to be the property corresponding to the proposition of satisfying both a and b, then clearly $c \leq a$ and $c \leq b$.
  - if d is a property corresponding to a logical proposition that implies both a and b, then clearly the proposition of satisfying d implies the proposition of satisfying both a and b. Hence, $d \leq c$.

The above explanations relied on the fact that we were working in the complete property space, because we assumed that the property corresponding to the intersection (or logical conjunction) *is present in* the property space. When working over a concrete property space that is not the complete space, we may not have such a guarantee.

Consider the following:

- Consider the space of all properties of words that can be tested in at most 10 seconds on a given computer. Clearly, the conjunction of two such properties may take upto 20 seconds to test on the given computer. Thus, the conjunction may not lie in the space.
- Consider the space of those properties that can be expressed using a single input output dependency. The conjunction of two such properties may involve two input output dependencies, hence it may not be in the space.

Meet can be considered, not just for two elements, but for an arbitrary collection of elements. The meet of a collection of elements a_w is the largest element a such that $a \leq a_w$ for all elements a_w in the collection. The meet may not be well defined for arbitrary collections. In fact, the meet being defined for every pair of elements only guarantees that meet is defined for finite collections.

A property space where the meet is defined for all subcollections is termed **arbitrarily conjunctive**(defined). Clearly, an arbitrarily conjunctive property space is conjunctive. The operation of taking the meet is termed **arbitrary conjunction**(defined).

The interpretation of arbitrary conjunction in terms of subcollections and logical propositions is analogous to that of conjunction for two elements.

2.2. **Disjunction.** The discussion for disjunction parallels that for conjunction, and is in some sense dual. It is given for the sake of completeness. However, despite the similarity of the notions, when we come to actually *applying* property theoretic notions, we shall see that conjunction and disjunction take on very distinct flavours.

The **join**(defined) of two elements in a partially ordered set is the smallest element that is bounded above by both of them. A poset where any two elements have a well defined join is termed a **join semilattice**(defined). The join is denoted by the symbol $\vee$.

When a property space is a join semilattice, the join operation is called **disjunction**(defined), and the property space is termed a **disjunctive property space**(defined).

For a complete property space, the disjunction operation can be understood by either of the two interpretations for properties seen so far:

- **As subcollections**: The disjunction of two properties corresponds to the subcollection that is the intersection of the subcollections corresponding to the two properties. The argument for this goes in an exactly analogous fashion to that for conjunction.
- **As logical propositions**: The disjunction of two properties corresponds to the *logical disjunction*, or the OR, of the corresponding propositions. The argument for this goes in an analogous fashion to that for disjunction.

For similar reasons to meet, the join operation may not be well defined for a general concrete property space, and even when defined, it may not correspond to the union of subcollections or the logical disjunction of the corresponding propositions. In fact, the same examples as used for meet serve to illustrate this:

- Consider the space of all properties of words that can be tested in at most 10 seconds on a given computer. Clearly, the disjunction of two such properties may take upto 20 seconds to test on the given computer. Thus, the disjunction may not lie in the space.
- Consider the space of those properties that can be expressed using a single input output dependency. The disjunction of two such properties may involve two input output dependencies, hence it may not be in the space.

Join can be considered, not just for two elements, but for an arbitrary collection of elements. The join of a collection of elements a_w is the largest element a such that $a \leq a_w$ for all elements a_w in the collection. The join may not be well defined for arbitrary collections. In fact, the join being defined for every pair of elements only guarantees that join is defined for finite collections.

A property space where the join is defined for all subcollections is termed **arbitrarily disjunctive**(defined). Clearly, an arbitrarily disjunctive property space is disjunctive. The operation of taking the join is termed **arbitrary disjunction**(defined).

The interpretation of arbitrary disjunction in terms of subcollections and logical propositions is analogous to that of disjunction for two elements.

2.3. **Property lattice.** A property space with both a conjunction and a disjunction is termed a **property lattice**(defined). This is because, when viewed as a partially ordered set, it becomes a lattice under these two operations.

If the property space is arbitrarily conjunctive and arbitrarily disjunctive, it is termed a **complete property lattice**(defined).

Any complete property space is naturally a complete property lattice, because the arbitrary conjunction and disjunction operations can be defined. Moreover, complete property spaces are also *completely distributive*, that is, the join and meet operations distribute over each other. We shall return to this point later on, when we encounter property spaces where distributivity breaks down.

2.4. **Negation and complementation.** In an abstract property space:

Definition. Two elements a and b are termed **complements**(defined) of each other if their disjunction is well defined and takes the value t, and their conjunction is well defined and takes the value f.

A **complemented property space**(defined) is a property space where every element has a complement. A **uniquely complemented property space**(defined) is a property space where every element has a unique complement.

When we are working over the complete property space, the subcollection and logical proposition viewpoint can again be used to give meaning to the complementation operation:

- **As subcollections**: If each property is identified with the subcollection satisfying it, then the complement of a property is simply the property whose associated subcollection is the complement of the subcollection satisfying the property. That is, if a and b are complements, then the subcollection satisfying a and the subcollection satisfying b are complements of each other. So, the elements satisfying b are precisely those that do *not* satisfy a.
- **As logical propositions**: The logical proposition corresponding to the complement of a property is simply the complement of the logical proposition corresponding to it. Thus, if p is a property, the logical proposition corresponding to p is $x \models p$ for an unknown element x in the context space, and that for the complement of p is $\neg(x \models p)$.

Just as for conjunction and disjunction, there are situations where we may not want a concrete property space to have a notion of negation. Here are two examples:

- Consider the **affirmative property space**(defined). A property is said to be **affirmative**(defined) if whenever an element of the context space is an instance of the property, a finite proof (in a fixed system of reasoning) of the fact can be furnished. The metaproperty of being affirmative is clearly a natural and important one. It is also true that a finite conjunction, as well as an arbitrary disjunction, of affirmative properties is affirmative. However, it is *not* true that if a property is affirmative then its complement must also be affirmative. This is because to check that there is *no* finite proof, we would in effect need to enumerate all possible finite proofs and see that they all fail.

  Issues related to affirmative property spaces shall be discussed in later articles.
- Consider the collection of all properties of executable programmes that can be specified by saying: "on inputs of length n, the programme will take at most $f(n)$ steps". Clearly, the complement of such a property will give a *lower bound* on the number of steps, and would not itself be a property of the form.

3. Property operators: basics

3.1. **Monotonicity and effect on extremes.** We begin with the study of property maps, that is, functions from one property space to another. Let S_1 and S_2 denote the two properties spaces, and $m : S_1 \to S_2$ denote the map. Then we say that m is:

- **monotone increasing**(defined) if $a \leq b$ implies $m(a) \leq m(b)$
- **monotone decreasing**(defined) if $a \leq b$ implies $m(b) \leq m(a)$
- **strictly monotone increasing**(defined) if $a < b$ implies $m(a) < m(b)$
- **strictly monotone decreasing**(defined) if $a < b$ implies $m(b) < m(a)$

In addition to monotonicity, another nice behaviour we would like in property maps is their effect on t and f. The monotonicity combined with the way the extremes are mapped, give rise to the **trace**(first used) of the property map. This notation is borrowed from existing notation used in some logic algebra systems. There are four possibilities:

- If m maps the tautology to tautology and is monotone increasing, it is said to have trace $+ \mapsto +$.
- If m maps tautology to fallacy and is monotone decreasing, it is said to have trace $+ \mapsto -$.
- If m maps fallacy to tautology and is monotone decreasing, it is said to have trace $- \mapsto +$.
- If m maps fallacy to fallacy and is monotone increasing, it is said to have trace $- \mapsto -$.

3.1.1. *From subgroups to groups.* Consider the following map h from the subgroup property space to the group property space. Given a subgroup property p, $h(p)$ refers to the group property whereby every non Abelian subgroup of the group has property p in the group. Clearly, h is monotone increasing – the stronger we make p, the stronger $h(p)$ becomes. It is not, however, strict monotone increasing.

The map h satisfies $h(t) = t$ but not $h(f) = f$. Thus, its trace is $+ \mapsto +$.

The statement that $h(t) = t$ follows from the formulation that *every* subgroup must satisfy the given property. In general, a formulation which says that *every* element somewhere must satisfy the input property, yields a map taking t to t. That is, the presence of *universal quantifiers* causes $h(t) = t$.

However, the property $h(f)$ simply means that every non Abelian subgroup of the given group must satisfy f. Because nothing can satisfy f, what it really means is that the group cannot have any non Abelian subgroups. In other words, $h(f)$ is simply the property of being Abelian. So $h(f) \neq f$.

3.1.2. *From groups to topological spaces.* Consider the following map τ from the group property space to the topological space property space. Given a group property p, let $\tau(p)$ denote the property of being a topological space whose fundamental group[2] has property p.

In this definition, we can easily see that $\tau(t) = t$ and $\tau(f) = f$. Further it is clearly monotone. Thus the trace of τ is both $+ \mapsto +$ and $- \mapsto -$. We will express this by saying that τ has trace $\pm \mapsto \pm$.

Unlike the previous example, there are no quantifiers.

3.1.3. *Intersection element.* Given a subgroup property p, let $u(p)$ denote the property of being the intersection of two normal subgroups with property p. Clearly, $u(f) = f$, but $u(t)$ is not the same as t. Thus the trace is $- \mapsto -$.

The underlying reason is that in this example, we make a statement of the form that there *exist* subgroups satisfying p, rather than asserting it for *every* subgroup.

3.1.4. *A summary of trace determination.* These are heuristic guidelines. They can be made precise, but we do not require that here:

- When there are no negatives involved, that is, everything is in the positive form, the property map is monotone increasing.
- For monotone increasing properties , in the *absence of existential quantifiers*, that is, in situations where all quantifiers are universal, t must map to t. Thus, if $h(p)$ is defined as the property such that every object arising in a certain manner satisfies p, the property map h takes t to t.
- For monotone increasing properties , in the *absence of universal quantifiers*, that is, in situations where all quantifiers are existential, f must map to f. Thus, if $h(p)$ is defined as the property such that there is an object arising in some way that satisfies p, the property map h takes f to f.
- When there is a single overall negation, the property map is monotone decreasing.
- When there are positives and negatives combined via conjunctions and/or disjunctions, then the property map may be neither monotone decreasing nor monotone increasing.

3.2. **Product of property spaces.** Let's first define the product of abstract property spaces:

Definition. Let P_1 and P_2 be abstract property spaces. The **product space**(defined) of P_1 and P_2 is the space $P_1 \times P_2$ with its tautology as (t, t), its fallacy as (f, f) and its partial order as the coordinate wise partial order (that is, $(p_1, p_2) \leq (q_1, q_2)$ provided that $p_1 \leq q_1$ and $p_2 \leq q_2$. This can be extended to define a product of arbitrarily many property spaces.

This product is simply a product as partially ordered collections. However, if the original spaces P_1 and P_2 are *concrete* property spaces, then we would like $P_1 \times P_2$ to also be given a concrete property space structure. Indeed we can do that:

Definition. Let P_1 be a concrete property space with context space S_1 and P_2 be a concrete property space with context space S_2. Then, the **product space**(defined) is defined as $P_1 \times P_2$ treated as a concrete property space over $S_1 \times S_2$ such that a given property (p_1, p_2) is satisfied by an element (s_1, s_2) if and only if $s_1 \models p_1$ and $s_2 \models p_2$.

This definition can be extended to arbitrary products of concrete property spaces.

Though the definitions seem straightforward, there are plenty of subtleties associated with them. Here are some of them:

[2]The fundamental group of a topological space is a group obtained from the topological space. Its definition need not concern us here.

- The product space of two complete property spaces is not a complete property space.
- The product of two abstract property spaces might be interpreted over a context space other than the product of the context spaces. This is particularly true in the case of **context combinators**(first used) and augmentations, that we shall come to later.

3.3. **General property operators.** A property operator is a property map from the product of certain property spaces, to a property space. More formally:

Definition. A **property operator**(defined) $S_1 \times S_2 \times \dots S_m \to S$ is a map from the product of property spaces S_1, $S_2 \dots S_m$ to the property space S. Alternatively, it can be viewed as a map from the collection of all n tuples with the i^{th} entry in S_i, to S.

Because the product of property spaces is itself a property space, the property operator can be viewed as a property map from the product space. However, viewing it as maps from n tuples over the original property spaces has the advantage that we can try to study the isolated behaviour of each individual property space.

Each i from 1 to m corresponds to a **place**(explained) in the property operator. The property operator is said to be monotone increasing, monotone decreasing, etc et era, in a place if the following holds:

> Fix the values in all the other places. Then, the property map from the value in that place is itself monotone increasing, monotone decreasing, or so on.

3.4. **Determining the trace via definitional quantifiers.** We had seen some heuristics about trace determination for property maps. With the introduction of property operators as the multifaceted analogue of property maps, many of those ideas go through. The examples (discussed in the next section) will make it clear.

3.5. **What we have achieved.** In the last two sections, we have been busy setting the stage for the magic of property theory. On the face of it, the definitions so far – product spaces, property operators, conjunctions, disjunctions, and so on, seem routine stuff that could be done for general partially ordered collections.

With this foundation, we shall see that n ary property operators from a property space to itself can be used to express notions spanning across a large number of context spaces and property spaces. In fact, we'll define some operators for each of the property spaces we used as examples in the first section, and see how these operators help capture a lot of the things we do in these context spaces.

4. Property space with binary operation

4.1. **Residuation.** Perhaps the most useful concept in property theory is that of **residuation**(first used). The idea has already existed in the study of logical structures. We'll define the notion for abstract property spaces.

Definition. Let $h : A \to B$ be a map of property spaces with signature $- \mapsto -$. Then define the **residuation**(defined) over h as a map $g : B \to A$ where $g(p)$ is a q such that $h(s) \leq p \iff s \leq q$.

Claim. Let A and B be arbitrarily disjunctive property spaces. Let h be a property map with signature $- \mapsto -$ such that:

$$h(\bigvee_{w \in W} p_w) = \bigvee_{w \in W} h(p_w)$$

Then, h has a residuation.

Proof. Let p be a property in B. We define the collection S of properties such that:

$$S = \{x \in A | h(x) \leq p\}$$

Because h has signature $- \mapsto -$, S contains the element f and is thus nonempty. Now, consider the following:

$$h(\bigvee_{s \in S} s) = \bigvee_{s \in S} (h(s))$$

$$\implies h(\bigvee_{s \in S} s) \leq p$$

So let q be defined as the left hand side (that is, the disjunction of all elements in S). Clearly, we have that $s \leq q \implies h(s) \leq h(q) \implies h(s) \leq p$. Conversely, if $h(s) \leq p$, we have $s \in S$, and hence $s \leq q$ (because q is the disjunction of all elements in S. □

Note that in the proof, we very crucially used the fact that the two spaces were arbitrarily disjunctive, *and* that the property map was compatible with the disjunction. As we shall see, most cases of $- \mapsto -$ maps that arise in practice satisfy these conditions. In fact, when the properties are defined purely with *existential quantifiers*, the very manner of definition enforces things.

Another claim, whose proof follows directly, is as follows:

Claim. The residuation of an operator of trace $- \mapsto -$ has trace $+ \mapsto +$.

Proof. The residuation is clearly a monotone operator. To demonstrate its trace, we thus need to compute it at t. But the collection of all properties s such that $h(s) \leq t$ is the collection of *all* properties. In particular, their disjunction is the t of A. □

4.1.1. *From groups to topological spaces.* Earlier on, we had defined τ as a map from the group property space to the topological space property space as follows: $\tau(p)$ is the property of being a topological space whose fundamental group has property p. Note that this definition is quantifier free, and, in particular, has a signature of $- \mapsto -$.

If p_1 and p_2 are group properties, then $\tau(p_1 \vee p_2)$ stands for the property of being a topological space whose fundamental group has property $p_1 \vee p_2$. This further translates to the statement that the fundamental group has property p_1 or the fundamental group has property p_2, and hence, it can be rewritten as $\tau(p_1) \vee \tau(p_2)$.

In fact, the same reasoning can be extended to arbitrary disjunctions. Thus, the map τ satisfies all the conditions listed above for us to conclude that it has a residuation. What does the residuation mean?

The residuation of τ simply means that, given a topological space property p, we return the property of being a group such that *every* topological space with that as its fundamental group must have property p. Note that this residuation, now, has a *universal quantifier* but no existential quantifiers. We can see that the residuation has signature $+ \mapsto +$.

4.1.2. *From subgroups to groups.* We had earlier seen the Hamiltonian operator that takes in an element of the subgroup property space and returns an element of the group property space, as follows: for a subgroup property p, the group property returned will be that *every* subgroup of it has the property p. This operator does *not* have signature $- \mapsto -$, nor is it arbitrarily disjunctive. The notion of residuation does not make sense for it.

Similarly, the operator that takes a subgroup property and returns the property of being a group inside which that property is transitive is not of trace $- \mapsto -$. In fact, it is even worse: it is not even a monotone operator. Again, the notion of residuation does not make sense for this operator.

Let's look at a a slightly different operator from the subgroup property space to the group property space. Given a subgroup property p, define the associated group property as the property of being a group with a subgroup having property p in it. Now, *this* has trace $- \mapsto -$ and is also arbitrarily disjunctive, because of its *existential* nature. The residuation of this operator takes in a group property p and returns the property of being a subgroup in a group with property p. Again, we see that t goes to t under the residuation map.

4.2. **The property magma.** We now define the following basic concept:

Definition. A **property magma**(defined) is a property space equipped with a binary operation that has trace $(-,-) \mapsto -$.

Property magmas basically encode *property operators* from a property space to itself. I have used the word *property magma* because in algebra, the term *magma* is sometimes used to describe a set with a binary operation.

Here is a common type of property magma.

Definition. A property magma is termed a **quantale**(defined) if it is arbitrarily disjunctive in each place. That is, a property magma with property space S and operation $*$ is termed a quantale, if, for any fixed $s \in S$, the maps $x \mapsto x * s$ and $x \mapsto s * x$ are both arbitrarily disjunctive.

The term *quantale* is, I believe, used in other logical settings with the same meaning (with the property space replaced by a partially ordered set).

Let's now define residuation for a property magma:

Definition.

- The **left residuation**(defined) in a magma $(S, *)$ is a map $l_* : S \times S \to S$ where $l_*(p, q)$ is evaluated as follows. Consider the map $x \mapsto x * p$. Now, this map has trace $- \mapsto -$ and is arbitrarily disjunctive. So, it has a residuation. Look at the residuation at q. The value thus obtained is the value of $l_*(p, q)$. We shall denote the left residuation $l_*(p, q)$ as q/p.

  In other words, $l_*(p, q)$ is defined as the property r such that $x * p \leq q \iff x \leq r$.
- The **right residuation**(defined) in a magma $(S, *)$ is a map $r_* : S \times S \to S$ where $r_*(p, q)$ is evaluated as follows. Consider the map $x \mapsto p * x$. Now, this map has trace $- \mapsto -$ and is arbitrarily disjunctive. So, it has a residuation. Look at the residuation at q. The value thus obtained is the value of $r_*(p, q)$.

  In other words, $r_*(p, q)$ is defined as the property r such that $p * x \leq q \iff x \leq r$.

  We shall denote the right residuation of $r_*(p, q)$ as $p \backslash q$.
- A property magma is said to be **left residuated**(defined) if the left residuation operation is well defined.
- A property magma is said to be **right residuated**(defined) if the right residuation operation is well defined.
- A property magma is said to be **residuated**(defined) if it is both left and right residuated.

Now, it's clear, from the way we have defined things, that quantales are residuated property magmas, because of arbitrary disjunction in both variables. In fact, most of the structures we are going to deal with are going to be quantales. However, it still makes sense to look at the generality of residuated property magmas.

4.3. **Property monoids.** In algebra, a magma is termed a **semigroup**(defined) if the binary operation is associative, and a **monoid**(defined) if the binary operation is associative and has a neutral element. In the same vein, we define:

Definition.

- A **property semigroup**(defined) is a property magma whose binary operation is associative.
- A **property monoid**(defined) is a property semigroup with a neutral element.

All the property magmas that we'll study are not going to be property monoids. The property monoids are basically some particularly *well behaved* property magmas.

Note that because of the $(-, -) \mapsto -$ signature, $f * x = x * f = f$ for all x. Thus, in particular, the property magma has a nil element f, which is hence also an idempotent. In particular, it cannot be cancellative and hence is pretty far from ever being a group.

4.4. **Elemental operators to property operators.** Suppose $f : S_1 \times S_2 \times S_3 \ldots S_n \to S$ is a function. Let P_i denote the property space with context space S_i, and P denote the property space with context space S. Then, we can define a map $\tilde{f} : P_1 \times P_2 \times \ldots P_n \to P$ as follows: $\tilde{f}(p_1, p_2 \ldots p_n)$ is the property of being an element in S that can be written as $f(s_1, s_2 \ldots s_n)$ where each s_i satisfies p_i.

The idea is that a function on the context space, viz an operator on the *elements* of the context space, gets lifted to an operator on the *properties* of the context space. This lifting can also be thought of as arising from the identification of the property with the subcollection satisfying it.

A property operator arising from an elemental operator in this way always has trace $(-, -) \mapsto -$, and in fact, gives a quantale structure. This is easy to see.

Here are some readily verifiable facts:

- If the operator on elements is commutative, so is the corresponding property operator.
- If the elemental operator is associative, so is the corresponding property operator.
- If the elemental operator has an identity element, so does the corresponding property operator: the property of being that identity element.

In fact, even *partially defined* elemental operators lift to *fully defined* property operators.

Here, we need to be a little more careful:

- If the partially defined operator is commutative whenever *either* side is defined, then the corresponding property operator is commutative.
- If the partially defined operator is associative whenever *either* side is defined, then the corresponding property operator is associative.
- If we have $a * (b * c) = (a * b) * c$ whenever the left side is defined for the partially defined operator $*$, we have, for the corresponding property operator $\tilde{*}$:

$$a \tilde{*} (b \tilde{*} c) \leq (a \tilde{*} b) \tilde{*} c$$

- If we have $a * (b * c) = (a * b) * c$ whenever the right side is defined for the partially defined operator $*$, we have, for the corresponding property operator $\tilde{*}$:

$$(a \tilde{*} b) \tilde{*} c \leq a \tilde{*} (b \tilde{*} c)$$

- If a partially defined binary elemental operator has, for every a, an e such that $a * e = a$, and such that $b * e = b$ whenever defined, then the property of being one such e is a right neutral element for the corresponding property operator. We analogously have the notion of left neutral element. (This concept is somewhat involved. We'll discuss it in another article).

In addition to elemental operators, we can also use "point to set maps" to give property operators. Point to set maps, or multivalued operators, are operators that can take on multiple values. Formally, a point to set map on S is a map $S \times S \to 2^S$. Given a point to set map, or a multivalued elemental operator$*$, the corresponding property operator $\tilde{*}$ is again defined as follows: $p_1 \tilde{*} p_2$ is the property of being an element in the set $a * b$ where $a \models p_1$ and $b \models p_2$. We again have corresponding results to those mentioned above (regarding associativity, commutativity and identity elements).

4.5. **Some illustrations.** In this subsection, we provide illustrations of binary operations defined on the property spaces we have been using for illustrative purposes. Some of these, as we shall see, turn them into quantales, which are occasionally associative and occasionally give monoidal structures

4.5.1. *Groups.* Some commonly encountered property operators on group properties are:

(1) **Direct product**(defined): Given two group properties p_1 and p_2 the direct product $p_1 \times p_2$ is defined as the property of being a group that can be expressed as a direct product of groups with properties p_1 and p_2 respectively.

The direct product operator on group properties is simply the property operator corresponding to the *elemental operator*, namely the direct product operator on isomorphism classes of groups. This elemental operator is commutative and associative, and has an identity element, namely, the trivial group. Hence, the associated property operator is also commutative and associative and has an identity element, namely the *property of being the trivial group.*

(2) **Extension**(defined): Given two group properties p_1 and p_2, define $p_1 * p_2$ as the property of being a group with a normal subgroup with property p_1 and such that its quotient group has property p_2. This operator is forward associative. This follows from the following fact: if N is a normal subgroup of G and N_1 is a normal subgroup of G/N, the inverse image of N_1 under the quotient map is again a normal subgroup of G. If we call this N_2, we also have $G/N_2 \cong (G/N)/N_1$.

However, reverse associativity does not hold because if N is a normal subgroup and M is a normal subgroup of N, M may not be a normal subgroup of the whole group.

The first of these gives a commutative property monoid, while the second one gives a property magma that is not even associative. However, it has one way associativity. As we shall get to see, many of the results that we develop are valid over structures with one way associativity.

4.5.2. *Subgroups.* In sections 1.6.2 and 1.7.2 we discussed some properties of subgroups, and some metaproperties.

Some commonly performed operations on subgroup properties are:

(1) **Composition**(defined): If p_1 and p_2 are two subgroup properties, $p_1 * p_2$ is defined as the following property: H satisfies $p_1 * p_2$ in G if there is an intermediate subgroup K (with $H \leq K \leq G$) such that H satisfies p_1 in K and K satisfies p_2 in G.

Let us define the composition of two subgroups $H \leq K$ and $L \leq G$ as $H \leq G$ provided that K is the same as L. This is a *partially defined* elemental operator, defined only when the bigger group of the left operand is the smaller group of the second. The corresponding property operator is, indeed, the above described composition operator.

Since the elemental operator is a partially defined operator and is associative whenever the expression on either side is defined, the property operator is also associative.

The property operator also has an identity element − the property of a subgroup being the whole group, that is, the property of the subgroup being the *improper* subgroup.

(2) **Lower meet**(defined): The lower meet of properties p_1 and p_2 is the property of being a subgroup obtained as the intersection of subgroups having properties p_1 and p_2 in the same bigger group.

Let us define the intersection of two subgroups $H \leq G$ and $K \leq L$ as $H \cap K \leq G$ if G is the same as L. This is again a *partially defined* elemental operator, with the definition making sense only if the bigger groups are the same in both cases.

The elemental operator is a partial operator that is commutative and associative whenever defined. Hence, the property operator is both commutative and associative.

The property operator has an identity element – the property of a subgroup being the whole group.

(3) **Lower join**(defined): The lower join of properties p_1 and p_2 is the property of being a subgroup obtained as the join of subgroups having properties p_1 and p_2 in the same bigger group.

Lower join also arises from an elemental operator, but it is a little tricky to analyze. It is a commutative and associative operator and has an identity element, namely the property of being the trivial subgroup. We shall not refer to it much in this article.

(4) **Upper join**(defined): The upper join of properties p_1 and p_2 is defined as the property of a subgroup H in a group G such that there are groups G_1 and G_2 between H and G such that G_1 and G_2 generate G, and H satisfies property p_1 in G_1 and p_2 in G_2.

This is somewhat different in flavour because it does *not* arise directly from an elemental operator. It is commutative and associative and an identity element for it is the property of being the whole subgroup. We shall not refer to it further in this article.

We shall use the following notation:

Operator name	Operator Symbol
Composition	$*$
Lower meet	$\cap_l$
Lower join	$\cup_l$
Upper meet	$\cup_u$

Notice that in *all* the above examples, the property operator was associative, so in fact the property magma is a property semigroup. Also in all the cases, there was also an identity element, so that the property magma is in fact a property monoid.

Plenty of elementary group theoretic results can be stated using these. For instance, the statement that a characteristic subgroup of a normal subgroup is normal boils down to saying that the composite of characteristicity and normality implies normality.

4.5.3. *Graphs.* Some operations over the property space of graphs:

- **Graph union operator**: Let p_1 and p_2 be two graph properties. Then define $p_1 \cup p_2$ to be the property of being a graph $G = (V, E)$ such that we have subgraphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ with $V = V_1 \cup V_2$ and $E = E_1 \cup E_2$, such that $G_1 \models p_1$ and $G_2 \models p_2$.

  There is no elemental operator directly corresponding to it, but we can associate a multivalued elemental operator. The property operator is clearly commutative and associative, and its identity element is the empty graph.
- **Edge partition operator**: Given two graph properties p_1 and p_2, the edge partition product $p_1 \sqcup p_2$ is the property of being a graph (V, E) such that there is a partition of the edges $E = E_1 \sqcup E_2$, we have both $(V, E_1) \models p_1$ and $(V, E_2) \models p_2$.

  The edge partition operator does not arise from an elemental operator. It is commutative and associative. The tautology t is the identity element for this operator.

  The edge partition operator is notoriously difficult to compute even for basic graph properties. In fact, the Ramsey problem can be thought of as a special case of this. $R(m, n)$ is the smallest number s such that there is no graph of cardinality s satisfying $p_m \sqcup p_n$ where p_m is the property of not containing a K_m and p_n is the property of not containing a K_n.

4.5.4. *Formal languages.* The property space over formal languages also has some naturally defined operations. Here are some of them:

- **Intersection**: The intersection of two properties is defined as the property of being a language obtained by intersecting languages with the two properties respectively, both over the same letter set. For instance, the intersection of property p_1 and p_2 is the property of being the intersection of two languages L_1 and L_2 both of which are over the same letter set.

  This is a property operator arising from a partial elemental operator – the operator of intersecting two languages over the same letter set. This operator is commutative and associative wherever defined, and hence, the property operator is commutative and associative.

  The property operator also has an identity element – the property of being the whole language over its letter set.
- **Union**: The union of two properties is defined as the property of being a language obtained by taking union of languages with the two properties respectively, both over the same letter set. For

instance, the union of property p_1 and p_2 is the property of being the union of two languages L_1 and L_2 both of which are over the same letter set.

This property operator is again both commutative and associative, because it arises from a partially defined elemental operator. It also has an identity element – the property of being the empty language.

- **Concatenation**: The concatenation of two properties is defined as the property of being a language obtained by concatenating a language with the first property, with a language with the second property. (This in turn depends on the definition of concatenation of languages as the language comprising words that can be split with a first part in the first language and the second part in the second language).

  The elemental operator here is the concatenation operator over individual languages, when both are defined over them same letter set. This elemental operator is associative, and hence, so is the property operator. The identity element here is the property of being the language comprising precisely one word – the empty word (note that this differs from the empty language).

We shall follow these conventions for the property operators:

Property operator	Property symbol
Union	$\cup$
Intersection	$\cap$
Concatenation	.

4.5.5. *Topological spaces.* Here are a few common operations in the topological space property space:

- **Product space operator**: Given two topological space properties p_1 and p_2, the property $p_1 \times p_2$ is defined as the property of being a topological space that can be expressed as a product of a topological space with property p_1 and a topological space with property p_2.

  This operator is commutative and associative, and also has an identity element – the property of being a single point space. It arises from the elemental operator of the product space of topological spaces, which is commutative, associative, and has an identity element – the property of being the one point space.

- **Fiber product operator**: Given two topological space properties p_1 and p_2, define their fiber product as the property of being a topological space E such that $p : E \to B$ is a fiber map with the fiber spaces F having property p_2 and the base space B having property p_1.

  Since every product can be treated as a fiber product, the product of two fibers is always stronger than the fiber product. The fiber product is not commutative or associative, but it does have an identity element – the property of being the one point space.

  We shall not look at the fiber product example further within this text.

5. Property magmas: in the thick

5.1. **Associativity.** Here's a quick recall of important definitions related to property magmas:

- A property magma is said to be **forward associative**(defined) if the following holds for all a,b and c.

$$a * (b * c) \leq (a * b) * c$$

- A property magma is said to be **reverse associative**(defined) if the following holds for all a,b and c.

$$(a * b) * c \leq a * (b * c)$$

- A property magma is said to be **associative**(defined) if the following holds for all a,b and c.

$$(a * b) * c = a * (b * c)$$

- Suppose $*$ and . are two property operators on a property space (both monotone positive with f going to f, as usual). Then we are interested in situations of $*$ being associative over ., such as

$$a * (b * c) = (a.b) * c$$

  and so on.

A property semigroup, as we had seen, was a property magma where the binary operation is associative.

In this section, we develop the theory of property magmas in general, with the minimum of additional assumptions. That is, our results are going to be applicable to all abstract property magmas, and hence can be interpreted for *each* of the concrete structures described above. And the beauty is that these common concerns interpret in different but related ways for each of the concrete structures.

5.2. **Some general definitions.** For this and the next two subsections, we assume that the property magmas are equipped with the required residuations. That is, whenever we talk of left residual, we assume that the property magma is left residuated, and when we talk of right residual, we assume that the property magma is right residuated. The square of a property a in a property magma is $a * a$. We begin with three definitions, for a property magma:

Definition.
- A property is said to be **transitive**(defined) if it is implied by its square. That is, a is transitive if and only if $a * a \leq a$.
- A property is said to be **subidempotent**(defined) if it implies its square. That is, a is subidempotent if and only if $a \leq a * a$.
- A property is said to be **idempotent**(defined) if it equals its square. That is, a is idempotent if and only if $a * a = a$.

Here is a little observation of interest:

Claim. Consider the property operator corresponding to a given elemental operator. If all elements satisfying a given property are idempotents, then the property operator is subidempotent. In particular, if the elemental operator is an idempotent operator, the corresponding property operator is subidempotent.

Here are some other definitions based on the relative nature with properties that are neutral elements (multiplicative identities) of the magma:

Definition.
- A property is said to be **left** i **true**(defined) if it is implied by a left neutral element.
- A property is said to be **right** i **true**(defined) if it is implied by a right neutral element.
- A property is said to be i **true**(defined) if it is implied by the neutral element.
- A property is said to be **left** i **strong**(defined) if it implies a left neutral element.
- A property is said to be **right** i **strong**(defined) if it implies a right neutral element.
- A property is said to be i **strong**(defined) if it implies the neutral element.

Note that any left or right i true property is subidempotent.

Given a property magma with a neutral element, a property that is both transitive and i true is termed a *t.i.* property. All *t.i.* properties are idempotent, and all idempotent properties are transitive.

Note that being transitive, i true and so on are all *metaproperties.*

Upshot. Given a property space, and a binary operation on it of trace $(-,-) \mapsto -$, we get some metaproperties on the property space, namely: the metaproperties of being transitive, idempotent, subidempotent, (left/right) i strong, (left/right) i true, with respect to the given operation.

We'll now look at the list of property spaces we have seen and the list of property operators defined over these, and see which properties are transitive, idempotent, i true etc. with respect to those property operators.

5.2.1. *Groups.*
- **Direct product operator**: This operator was commutative, associative, and with an identity element – the property of being the trivial group.
  - A group property is i true with respect to this operator if it is satisfied by the trivial group. Clearly, the properties of being cyclic, Abelian, solvable, perfect, finite, trivial are all i true. On the other hand, the property of being infinite is not i true.
  - A group property is transitive with respect to this operator if the direct product of any two groups with the property also has the property. The property of being Abelian, solvable, finite, trivial, perfect, infinite, are all transitive, but the property of being cyclic is not.
- **Extension operator**: This operator was forward associative, with a two way identity element – the property of being the trivial group.
  - The notion of i true remains the same as for direct product
  - The notion of transitive with respect to this operator is stronger than the corresponding notion for direct products. The properties of being finite and being solvable are transitive with respect to this operator.

5.2.2. *Subgroups.*

- **Composition operator**:
  - A property of subgroups is *transitive* with respect to the composition operator if it is transitive in the sense described in section 1.7.2. That is, if p is a property of subgroups, p is transitive if whenever $H \leq K$ has property p, and $K \leq G$ has property p, so does $H \leq G$. Of the subgroup properties we saw, the following are transitive: whole group, characteristic, fully invariant central factor, retract, Sylow subgroup, trivial subgroup. The property of being normal is *not* transitive.
  - A property of subgroups is i true with respect to the composition operator if the every group has the property as a subgroup of itself. Some subgroup properties that are i true: the properties of being the whole group, characteristic, fully invariant, central factor, retract, normal. In fact, all of them, except the property of being normal, are *t.i.*.
  - All i true properties are subidempotent, and hence, all the properties we mentioned above are subidempotent. With the exception of normality, they are all idempotent. The properties of being the Sylow subgroup, and also the property of being the trivial subgroup, are idempotent, even though they are *not* i true.
- **Lower meet operator**:
  - A property of subgroups is *transitive* with respect to the lower meet operator if the intersection of two subgroups with the property in a bigger group also has the property. The following subgroup properties are transitive with respect to the lower meet operator: whole group, characteristic, fully invariant, normal subgroup, trivial subgroup.
  - The notion of i true remains the same as in the case of the composition operator, because the identity is the same: the property of being the whole group.

5.2.3. *Graphs.* Here are some observations regarding operators on the simple graph property space:

- **Graph union operator**:
  - A property of graphs is *transitive* with respect to this operator if given any property p, any graph that is a union of two graphs each with property p, also has property p. The properties of being finite, locally finite and empty are transitive with respect to this operation.
  - A property of graphs is i true with respect to the operator if it i s true for the empty graph. Clearly, the properties of being finite, connected, locally finite, regular, triangulable, are all i true.
- **Edge partition operator**:
  - A property of graphs is *transitive* with respect to the operator if given any property p, any graph whose edges have a partition into two parts both with property p, the whole graph also has property p. The property of being regular is transitive, because any graph whose edges can be partitioned into two parts with degrees d_1 and d_2 respectively, must have degree $d_1 + d_2$. The properties of being connected, finite, triangulable, are all transitive.
  - A property of graphs is i true with respect to the operator if it is true for every graph with no edges. Thus, the properties of being regular, triangulable, disconnected, are all i true.
  - Thus, two *t.i.* properties with respect to this operator are the properties of being triangulable and being regular.

5.2.4. *Formal languages.*

- **Intersection operator**:
  - *All* properties are subidempotent. This is because the associated elemental operator is idempotent.
  - A property of formal languages is *transitive* with respect to the intersection operation if and only if the intersection of any two languages with the property also has the property. Among the formal language properties we studied, the following are transitive: finite, star free, regular, whole language, and recursively enumerable. The property of being nonempty is not transitive because the empty language can be expressed as an intersection of two nonempty languages.
  - The identity here is the property of being the *whole* language on the letter set. Thus, of the formal language properties we have studied, the only one that is i strong is the identity itself. The i true properties include the properties of being star free, regular, context free, recursively enumerable, nonempty. Note that the property of being finite is *not* i true.
- **Union operator**:

  - *All* properties are subidempotent. This is because the associated elemental operator is idempotent.
  - A property of formal languages is *transitive* with respect to the union operation if and only if the union of any two languages with the property also has the property. The properties of being finite, star free, regular, context free, recursively enumerable, infinite, are all transitive with respect to unions. However, the property of being *deterministic context free*, that is, recognizable by a deterministic push down automaton, is not transitive with respect to unions. That is, it is not necessary that a union of deterministic context free languages be regular.
- **Concatenation operator**:
  - A property is transitive with respect to the concatenation operator if concatenating any two languages with the property also gives a language with the property. All the properties that we saw are transitive with respect to the concatenation operator: finite, star free, regular, context free, whole language, recursively enumerable.
  - A property is i true with respect to the concatenation operator if the formal language comprising precisely the empty word, satisfies the property. All the properties that we saw, except the property of being the whole language, is i true. That is, the following are i true: finite, star free, context free, regular, nonempty, recursively enumerable.
  - Hence, the properties of being finite, star free, context free, regular, recursively enumerable, are all *t.i.* with respect to the concatenation operator.
  - The following property is transitive with respect to the concatenation operator, but not *t.i.*: The property of the language containing at least one nonempty word. Clearly, the language comprising precisely the empty word does not satisfy this property.

5.2.5. *Topological spaces.* Here is a quick analysis of the property operators on topological spaces:

- **Product space operator**:
  - A property is *transitive* with respect to this operator if the product of any two spaces with the property also has the property, when given the product topology. Among the properties we had seen, the following are transitive: the properties of being Hausdorff, regular, connected, compact, metrizable. The property of being paracompact is *not* transitive, because there is a paracompact space whose product with itself is not paracompact. Similarly, the product of two normal spaces need not be normal.
  - A property is i true with respect to this operator if it is true for the one point space. Clearly, the properties of being Hausdorff, regular, connected, compact, metrizable, normal, Baire space, paracompact, are all i true.
- **Fiber product operator**

5.3. **Basic results.** We now look at some actual results for property magmas. As mentioned at the beginning of the previous subsection, these results are *not* specific to a particular property magma, and hence can be interpreted for practically all the examples.

Because of the left right symmetry in many of these results, the results are stated with one convention. The dual statement can be obtained by replacing the phrases by the corresponding phrases in the brackets, simultaneously in the whole statement.

Recall that a is stronger than b or implies b if $a \leq b$, and a is weaker than b if $b \leq a$.

Theorem 1 (Residuation Master Theorem)**.**

- If, in a property magma, $p_l * p_r = q$, then the right (respectively left) residuation of q by p_r (respectively p_l) is exact, that is, $p_l * p_l \backslash q = q$ (respectively $q/p_r * p_r = q$).
- , In a property magma with a left (respectively right) neutral element, if $p \leq q$, the left (respectively right) residual of q by p is left (respectively right) i true.
- In a forward (respectively reverse) associative property magma, if $p \leq q$ are properties, the right (respectively left) residuation of p by q is a transitive property.
- If p is transitive and $p \leq q$, then the right (respectively left) residuation of q by p is weaker than p.
- if p is left (respectively right) i true, then the right (respectively left) residuation of q by p is stronger than q.

Each of these results has a very direct proof.

When the property magma is commutative, we can simply talk of *the* residuation $\frac{q}{p}$ without any qualification via left and right.

5.4. **Transiters: some results.** The **minimum left transiter**(defined) of a is the left residuation of a by a, and is to be denoted as a/a. Note that the word *minimum* here means *weakest*, which, in the partial order, is actually the maximal element.

The **minimum right transiter**(defined) of a is the right residuation of a by a, and is to be denoted as $a\backslash a$.

Again, when the property magma is commutative, we can simply talk of *the* **minimum transiter**(defined) of a, which we shall denote by $\frac{a}{a}$.

Any property that is stronger than a minimum transiter is simply termed a **transiter**(defined). Analogously, we have the notion of left and right transiter.

The following results on transiters follow directly from theorem 1:

Theorem 2 (Transiter Master Theorem)**.**

- In a magma with left (respectively right) neutral element, the minimum left (respectively right) transiter of an element is left (respectively right) i true.
- In a forward (respectively reverse) associative property magma, the minimum right (respectively left) transiter of an element is itself transitive.
- A transitive property is stronger than both its minimum left and its minimum right transiters.
- A left (respectively right) i true property is weaker than its minimum right (respectively left) transiter.

When the property magma has a neutral element, the above simplifies to:

Theorem 3 (Simplified Transiter Master Theorem)**.**

- In a property magma with neutral element, any *t.i.* (transitive and i true) property is its own minimum left transiter, and its own minimum right transiter.
- When the property magma has forward (respectively reverse) associativity, the minimum right (respectively left) transiter of any element is *t.i.*.

The upshot of this is the idempotence theorem for property monoids:

Theorem 4 (Transiter Idempotence for property monoids)**.** The minimum left transiter and minimum right transiter of any property are *t.i.*, and the minimum left and right transiter of any *t.i.* property are itself. Thus, both the minimum left transiter map and the minimum right transiter map are idempotent maps, with the same fixed point collection. This fixed point collection is the space of all *t.i.* properties.

5.5. **Reviewing the above by some examples.** In the last two subsections, we saw a lot of results that, although obvious in the abstract context, yield important insights in every concrete case where we interpret it. This does not mean that the insight we gain could not have been achieved by separately arguing in each case. But the approach of treating a property magma definitely gives a new twist.

5.5.1. *Groups.* The direct product operator, that we have already studied, has been shown to give a commutative associative property monoid structure. The i true properties include things like being finite, Abelian, solvable, nilpotent, cyclic, perfect, and so on. The properties of being finite, infinite, Abelian, solvable, nilpotent are all transitive with respect to this operator. The property of being infinite is transitive but not i true, and the property of being cyclic is i true but not transitive.

Let's try to compute the transiter of the property of being cyclic. The question is: what property must an abstract group have so that its product with every cyclic group is cyclic? Clearly, because the property of being cyclic is i true, its minimum transiter must be at least as strong as it.

However, we can readily see that if G is a cyclic group, then $G \times G$ is not cyclic unless G is trivial. Thus, the *only* group whose direct product with every cyclic group is cyclic, is the trivial group. Thus, the minimum transiter of the property of being cyclic is the identity element (the property of being the trivial group).

What about the property of being infinite? This property is transitive but not i true. Hence, by the Transiter Master Theorem, its minimum transiter is weaker than it. In fact, we can see that the direct product of an infinite group with any group is infinite. Hence, the minimum transiter of the property of being infinite is the tautology t, the property held by all groups.

Let's now shift focus to the extension operator. The extension operator is a bit different from the other operators because it is not associative, it is only forward associative. Thus, the idempotence theorem for property monoids, and even the simplified form of the transiter master theorem is not available to us. We need to work with the general form of the transiter master theorem.

The properties of being solvable, finite, trivial are closed under extension. However, the properties of being cyclic, Abelian, are not. Clearly the minimum left and right transiter with respect to this operation are certainly stronger than the minimum transiter with respect to the direct product operator. Moreover, they are both i true. Because the minimum transiter for the property of being cyclic (with respect to the direct product operator) is anyway e, we conclude that the minimum left and right transiter with respect to extension are both e.

What about the property of being Abelian? What should a group be such that the extension of any Abelian group by it is also Abelian?

5.5.2. *Subgroups.* Let's first look at the composition operation. We have made the following observations about it so far:

- The composition operator arises from a partially defined elemental binary operator. Since that elemental operator is associative whenever either side is defined, and possesses elements that uniformly behave as left and right identities whenever multiplication is defined, the corresponding property magma is a property monoid.
- The following are all i true properties in the monoid: normal, characteristic, fully invariant, central factor, whole group,. Apart from the property of being normal, all are transitive and hence also *t.i.*.

Let's first see what the idempotence theorem for property monoids tells us. Clearly, all the properties listed except normality, being *t.i.*, are their own minimum left, as well as minimum right transiters.

What about the property of being normal? We already know that the minimum left transiter of normality is *t.i.*. Let's try to unfold the meaning of the statement.

> Let σ denote the minimum left transiter of normality. Then σ is such that H has the property σ as a subgroup of G, if and only if, whenever there is a group K with G as a normal subgroup of K, H is also a normal subgroup of K.

In other words, if H has the property σ in G, and G is normal in K, H is also normal in K. That is,

$$\sigma * \text{normal} \leq \text{normal}$$

We also require σ to be *minimal* for this – that is, we declare H to have property σ in G whenever it is true that whatever bigger group K we have with $G \trianglelefteq K$, we also have $H \trianglelefteq K$.

Let's go back to the transiter master theorem. This states that an i true property is always strengthened by its minimum left and right transiters. In other words, the minimum left transiter of normality must be at least as strong as normality. Can we see this directly? Yes. Simply take K to be equal to G with the identity embedding. Clearly, G is normal in K, and hence, H is normal in K, and hence H is normal in G.

In fact, the minimum left transiter of normality turns out to be the property of being characteristic. The minimum right transiter has no name of its own, but it is easy to see that it is characterized as follows: "every normal subgroup of the subgroup must be normal in the whole group".

The problem of transiter computation will be discussed in a later article, titled "tools in property theory".

What about the property of being a Sylow subgroup?

This property is transitive but not i true, hence, by applying the transiter master theorem, its minimum left and minimum right transiters must be *weaker* than it. What really is the minimum right transiter? It can be described as follows:

> Let σ denote the minimum right transiter of the property of being a Sylow subgroup. Then H has property σ in G if every Sylow subgroup of H is also a Sylow subgroup of G.

With the use of Sylow's theorem and some other basic ideas, it is fairly easy to compute this minimum right transiter. It comes out as the property of being a **Hall subgroup**(defined): a subgroup whose order and index are relatively prime. Clearly, the property of being a Hall subgroup is *t.i.*, and every Sylow subgroup of a Hall subgroup is also a Sylow subgroup of the whole group. The reverse claim (that is, Hall subgroups are precisely the minimum right transiter of normality) follows by applying Sylow's theorem.

The minimum left transiter is defined as follows: "if the bigger group has prime power order, then the subgroup is either trivial or the whole group. Otherwise, it could be any subgroup".

Now for the residuation situation in this property monoid. Here are a couple of questions:

- What is the property σ that a subgroup H must have in a group G such that every normal subgroup of H is a characteristic subgroup of G?

  Observe that normality is i true but not transitive, and characteristicity is *t.i.*. Also, the property of being characteristic is stronger than the property of being normal. Applying the residuation master theorem, we obtain that the property is stronger than that of being characteristic, and also that it is transitive. Clearly, it is not i true. Here are some properties stronger than the given property: the property of being the trivial subgroup, the property of being a minimal characteristic subgroup.
- What is the property σ that a subgroup H must have in a group G such that every characteristic subgroup of H is normal in G?

  Again, we begin with the same observations. We readily see from the residuation master theorem that the property is i true, and that it is at least as strong as the property of normality. Here are some properties that are stronger than it: the property of being a minimal normal subgroup, the property of being a subgroup inside the center.

Let's now look at the lower meet operator. First of all, note that this gives a commutative subidempotent property monoid, so we can simply talk of *the* transiter.

Recall that the properties of normality, characteristicity, whole group, full invariance, are all transitive (and in fact idempotent). However, of these, the property of being a Sylow subgroup is not i true or transitive.

So we have the question: what is the minimum transiter of the Sylow property with respect to lower meet? That is, what is the property of being a subgroup whose intersection with every Sylow subgroup is a Sylow subgroup? The property comes out as that of being a normal Hall subgroup.

5.5.3. *Formal languages.* Let's first look at the insights we can get with respect to the intersection operator.

We know that the main property of formal languages, that is important and not closed under intersections, is the property of being context free. Here, the operations are commutative, so a sensible question might be: what is the minimum transiter of the property of being context free?

There is an elementary result:

> The intersection of a context free language and a regular language is context free.

This tells us that the minimum transiter for context free languages is a property that is at most as strong as that of being regular and at least as strong as that of being context free. Where in between does it lie? An interesting question. In fact, there are context free languages whose intersection with every context free language is context free, even though they are not regular. Hence, the property is strictly in between the properties of regular and context free.

Now let's look for the application of the residuation master theorem. Here are some questions (that can be approached using the various provisions of the residuation master theorem):

- What is the transiter of regular languages by context free languages? That is, what languages have the property that their intersection with every context free language is regular?

  Clearly, this is the situation where we can apply the third provision of the residuation master theorem, in other words, the property, whatever it is, must be transitive. That is, if p is the property of being a language whose intersection with every language is context free, then the intersection of two languages with property p also has property p.
- What is the property of being a language whose intersection with every regular language is context free?

  This is an instance for the second provision of the residuation master theorem, that is, we can safely say here that the property in question is i true. Actually, in this case, we can say that the property is precisely that of being context free – because regularity itself being i true, it cannot be weaker than being context free.

5.5.4. *Topological spaces.* Let's first look at the product space operator.

The property of being paracompact is *not* transitive with respect to this operation. This is because the product of two paracompact spaces need not be paracompact. The property of paracompactness is i true, so the minimum transiter must be a strengthening.

By an application of the tube lemma, we find that the product of a compact space and a paracompact space is paracompact. Thus, the minimum transiter of paracompactness is *weaker* than the property of being compact. Hence, it is somewhere in between the properties of paracompactness and compactness.

In fact, compactness is a transiter (in the sense of being stronger than a minimum transiter) for the following properties:

- Orthocompactness: The product of a compact space and an orthocompact space is orthocompact.
- Metacompactness: The product of a compact space and a metacompact space is metacompact.
- Lindelof: The product of a compact space and a Lindelof space is Lindelof.

All the proofs use the tube lemma.

The reverse question: is compactness the *minimum* transiter? I don't know the answer to that question.

6. Another operator and more transiters

6.1. **Kleene star operator.** The **Kleene star**(defined) or ***t.i.* closure**(defined) of an element (in a property magma with neutral element) is the strongest element that is weaker than it, and that is *t.i.*. The Kleene plus or transitive closure of an element is the strongest element that is weaker than it, and is transitive.

The Kleene star is sometimes called the identity true subordination and the Kleene plus, the proper subordination.

When the element is power associative, the Kleene star is the disjunction of all the powers. Otherwise it is the disjunction of all products with only that element, and any kind of parenthesization.

Thus, in particular, for a property monoid, we have the following expressions for Kleene star and Kleene plus:

$$\text{Kleene star of } a = a^* = e \vee a \vee a * a \vee a * a * a \vee a * a * a * a \dots$$
$$\text{Kleene plus of } a = a^+ = a \vee a * a \vee a * a * a \vee a * a * a * a \dots$$

When a is transitive, $a = a^+$ and when a is i true, $a^+ = a^*$. When a is *t.i.*, $a = a^*$.

A **Kleene star property monoid**(defined) is a property monoid where the Kleene star operator is defined for all properties. We now have the following result for property monoids.

Theorem 5 (Idempotence Theorem for Kleene star)**.** *In a Kleene star property monoid, the Kleene star of any property is a* t.i. *property. Conversely, a property that is* t.i.*is its own Kleene star. In other words, the Kleene star operator is idempotent with the fixed point cum image space being the space of* t.i. *properties.*

There is, of course, no guarantee that the Kleene star or Kleene plus will be defined for a given property magma or property monoid. A property magma (with neutral element) where it is defined is termed a **Kleene star property magma**(defined) Any property magma that is countably disjunctive is automatically Kleene star.

Theorem 6 (Monotonicity and related results)**.**

- The Kleene star operator is monotone and weakening. That is, $a \le a^*$ and if $a \le b$, $a^* \le b^*$. In particular, if $a \le b$ and b is *t.i.*, then $a^* \le b$.
- The Kleene plus operator is also monotone and weakening, and the result is a transitive property, though not necessarily *t.i.*.

6.2. **The (m, n) minimum transiter.** In this section, we assume we are dealing with residuated property monoids.

The minimum (m, n) transiter of a property a is defined as the minimum property x such that $a^m * x * a^n \le a$. The minimum left transiter is the minimum $(0, 1)$ transiter and the minimum right transiter is the minimum $(1, 0)$ transiter. The value $m + n$ is called the **total order** of the transiter operation. We have already studied the behaviour of both the order 1 transiter operations. Here is some higher order behaviour:

Theorem 7 (Higher order transiter theorem)**.** For property monoids:

- If a property is *t.i.*, it equals all its higher order transiters.
- If a property is transitive, its order 2 transiters are i true.
- If a property is i true, its order 2 transiters are transitive.
- Thus, the order $4k$ transiters of transitive properties are transitive, and the order $4k+2$ transiters are i true.

Both the order 2 results come by applying the transiter master theorem twice. The order 4 result comes by applying the order 2 result repeatedly.

These results shed more light on some of the examples we have already seen.

In the case of commutative property monoids, there is a unique minimum transiter of order n.

Here, now, is another powerful result for property monoids with subidempotence:

Theorem 8 (Subidempotence stability theorem)**.** In a property monoid with all elements subidempotent, the minimum $(0, n)$ transiters are the same for $n \geq 2$, and the minimum $(m, 0)$ transiters are the same for $m \geq 2$.

Proof. Suppose $x * a * a \leq a$. Then:

$$\begin{aligned} x &\leq x * x \\ \implies x * a * a &\leq x * x * a * a \\ \implies x * a * a &\leq x * (x * a * a) \\ \implies x * a * a &\leq x * a \end{aligned}$$

Thus, $x * a^n \leq x * a^{n-1}$ whenever $n \geq 2$. In particular, $x * a^n \leq x * a * a$ when $n \geq 2$. Hence, $x * a^n \leq a$ whenever $n \geq 2$.

The upshot:

$$x * a^2 \leq a \implies x * a^n \leq a \; \forall \; n \geq 2$$

Conversely, we know by subidempotence that $x * a * a \leq x * a * a * a$ and so on. In other words, we have:

$$x * a^n \leq a \text{ where } n \geq 2 \implies x * a^2 \leq a$$

Thus, we get:

$$x * a^2 \leq a \iff x * a^n \leq a \text{ where } n \geq 2$$

From which it follows that the minimum $(0, 2)$ transiter and the minimum $(0, n)$ transiter are the same. Similar reasoning works in the dual case of $(m, 0)$. □

The above proof seems somewhat involved, but it arose from a very special case, where the reasoning seems quite simple. The special case is where we are talking of subset properties on a set with additional structure, and the $*$ denotes intersection. Then, if a is a property, A is the whole set and B is a subset of A with a property x so that $x * a * a \leq a$, it means that the intersection of B with two subsets with property a also has property a.

We claim that the intersection of B with any finite number of subsets $C_1, C_2 \ldots C_n$ each with property a also has property a. Let D be the intersection of B, C_1 and C_2. Then D has property a. Also, $D = B \cap D$, and hence, the intersection of B, C_1, C_2 andC_3 is the intersection of B, D and C_3, which is the intersection of B with two subsets having property a. So this intersection again has property a. Call this D_1. Then, D_1 can be viewed as $B \cap D_1$ and hence the intersection of B, B_1, B_2, B_3 and B_4 is also the intersection of B, D_1 and B_4, which is again the intersection of B and two subsets with property a. SO this again has property a The argument goes on ad infinitum.

What we are really using in the above proof is that $B \cap B = B$. This fact, when translated to the property space, becomes the fact that the property x is subidempotent.

6.3. **The Kleene star transiter.** The minimum left (respectively right) Kleene star transiter is the left (respectively right) residuation by the Kleene star. For subidempotent property monoids, the above proof can be used to show that the left/right Kleene star transiter is the $(0, 2)$ or $(2, 0)$ transiter itself.

6.4. **Some examples again!** We now look at the various property spaces, this time with the view to understanding higher order transiters, as well as the Kleene star operator.

6.4.1. *Groups.* Let's look first at the direct product operator.

The property of being cyclic was not *t.i.* with respect to this operator. So the question might be: what is the Kleene star closure of this property with respect to direct products. That is, what are the groups that can be written as finite length direct products of cyclic groups? Clearly, any such group must be a finitely generated Abelian group. Moreover, the famous **structure theorem for finitely generated Abelian groups** tells us that every finitely generated Abelian group can be expressed this way. Thus the Kleene star closure of the property of being cyclic is the property of being finitely generated Abelian.

For the property of being infinite, as this property is transitive, its Kleene star is simply its disjunction with the identity element. That is, the Kleene star of the property of being infinite is the property of being infinite or trivial.

6.4.2. *Subgroups.* First, for the composition operator.

Among the important properties we had seen, normality was a property that was not transitive. We had seen that the minimum left transiter of normality is the property of being characteristic, and the minimum right transiter, while not explicitly studied before, has a definite characterization. What about the Kleene star closure?

The Kleene star closure of the property of normality is termed "subnormality". A subgroup H of a group G is said to be subnormal in G if there is a chain of subgroups $H = H_0 \leq H_1 \dots H_n = G$ such that $H_i \trianglelefteq H_{i+1}$. Clearly, this ties in with the view of subnormality being normal $\vee$ normal $*$ normal $\vee$ normal $*$ normal $*$ normal and so on.

What about the higher order transiters of normality? Actually, we dealt with the $(1, 1)$ transiter earlier on in the examples: the problem of determining the property of H in G such that every normal subgroup of H is characteristic in G. We had observed that this property is transitive, which now follows directly from the higher order transiter theorem.

If a property is already transitive, its Kleene star is simply its disjunction with the property of being the whole group.

6.4.3. *Graphs.* Let's look at the edge partition operator.

What is the Kleene star closure of the property of being cyclic? Clearly, this property is not transitive – a graph whose edge set can be partitioned into two cyclic graphs may not itself be cyclic. In fact, it will *not* be cyclic.

However, we know that every cyclic graph is regular of even degree. And the property of being regular with even degree is *t.i.*. Hence, the Kleene star closure of the property of being cyclic, is stronger than the property of being regular of even degree. Is it equal? The question can be translated to the following: is it true that every regular graph of even degree can be expressed as an edge disjoint union of cyclic graphs? The answer is definitely no. In fact, the graph may be disconnected. This suggests that perhaps we should consider "regular connected graph of even degree". However, there are regular connected graphs of even degree that are not Hamiltonian (checking for Hamiltonicity is a nontrivial problem) and hence, even the property of being "regular connected of even degree" is too strong to be the Kleene star closure of the property of being cyclic.

6.4.4. *Formal languages.* Let's look at the intersection operator.

The most important property that was not transitive with respect to the intersection operator was the property of being context free. The Kleene star of this property would, naturally, be the property of being a finite intersection of context free languages. This is also termed the property of being a **concurrently context free language**, or a CCFL.

What about the higher order transiters for being context free? Because of subidempotence, we know that all its transiters of order 2 or more are equal.

6.4.5. *Topological spaces.* Let's look at the product space operator.

Some of the properties that are not transitive with respect to the product space operator are the property of being paracompact, being normal, and so on. The Kleene star closure of the property of paracompactness is the property of being a space that can be expressed as a finite product of paracompact spaces. I don't know if there is a name for it. Similarly, the Kleene star closure of the property of normality is the property of being expressible as a finite product of normal spaces.

Index