Lattice Tutorial
Version 1.0

Nenad Jovanovic
Secure Systems Lab
www.seclab.tuwien.ac.at
enji@infosys.tuwien.ac.at

November 3, 2005

1 Introduction

This tutorial gives an introduction to a number of concepts of lattice theory, especially to those needed for understanding data flow analysis. It starts with the basics, requires only little previous knowledge, and tries to illustrate the topics with concrete examples. I felt the need for such a tutorial because other works in this area that I came across so far have at least one of the following drawbacks:

- They are too short and shallow.
- They are too complex, filled with cross-references, and not self-contained.
- They are too long and usually not available online (books).
- They are slideshows, which tend to be rather unappealing without additional explanations by a human speaker.
- The covered material deviates from the topics needed for data-flow analysis too strongly.

You can find the latest version of this tutorial under the following URL:

http://www.infosys.tuwien.ac.at/Staff/enji/lattice_tutorial.pdf

2 Sets

As already mentioned, only little previous knowledge is required to understand this tutorial. **Sets** belong to these things you need to know about - you probably learned this at school already. For the beginning, our ”running gag” will be the set of natural numbers from 1 to 3: {1, 2, 3}.

3 Binary Relations

A **binary relation** R over a set is quite simple: It takes two elements from this set as input and returns "true" or "false" as output. For instance, consider the relation "lesser than or equal to" ($\leq$) over the set {1,2,3}. If you give the input (1,2) to this relation, it returns "true" (since $1 \leq 2$). For (3,2), it returns "false".

4 Partial Order Relations

Partial order relations belong to the class of binary relations, but additionaly have the following special characteristics:

- Reflexivity: If we take an element a of the underlying set, then the relation for the pair (a,a) must be true. For instance, the pair (1, 1) returns "true" for the relation $\leq$. As a shorthand, we write "aRa" for "relation R returns 'true' for the input (a,a)".
- Antisymmetry: If aRb and bRa, then a must be equal to b. For example, $1 \leq x$ and $x \leq 1$ can only be true if $x = 1$.
- Transitivity: From aRb and bRc, it follows that aRc (e.g. $1 \leq 2$ and $2 \leq 3$ leads to $1 \leq 3$).

As we saw in the examples for reflexivity, antisymmetry and transitivity, the relation $\leq$ satisfies all these properties, and therefore is a partial order relation.

5 Partially Ordered Sets

A set together with a partial order relation is called **partially ordered set** (short name: **poset**). You might have wondered what the word "partial" is supposed to mean in this context. It means that there *does not need to be an order for all pairs* of elements from the underlying set. If there *is* an order for all pairs, we are dealing with a special form of partially ordered set, namely the **totally ordered set**. In fact, the example we were discussing so far (the set {1,2,3} with the relation $\leq$) is a totally ordered set. For demonstrating a partially ordered set which is not a totally ordered set, first consider the **powerset** of {1,2,3}. This is a set containing all subsets of {1,2,3}: { {1,2,3}, {1,2}, {1,3}, {2,3}, {1}, {2}, {3}, {} } Note that a powerset also includes the empty set, since the empty set is a subset of every set. The notation for the powerset of some set S is $\mathcal{P}(S)$ or 2^S. Now we define a partial order on this set, namely the subset inclusion $\subseteq$. Here are some pairs for which this relation is true:

({1}, {1,2})	because $\{1\} \subseteq \{1,2\}$
({2}, {1,2,3})	...
({2,3}, {1,2,3}	
({}, {1})	because the empty set is subset of every set

This powerset is not totally ordered, but only partially ordered. The reason is that it contains pairs elements of elements for which no order exists, such as:

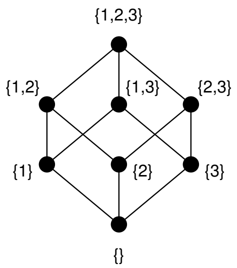

Figure 1: A Hasse diagram.

({1}, {2}) because neither {1} ⊆ {2} nor {2} ⊆ {1}
({1,2}, {2,3}) . . .

Now we have already seen two partial order relations: ≤ and ⊆. From now on, we will write $\sqsubseteq$ to denote some arbitrary partial order relation.

Posets can be depicted graphically. We can represent the poset elements as nodes and the relation between pairs of elements as directed edges (i.e. edges with an arrow at one end). However, this method is rather awkward because the number of edges increases rapidly with the number of elements. A more elegant way is to use a "Hasse diagram", which makes some of the explicit information implicit and hence becomes smaller. More precisely, it does the following:

- Reflexive edges (i.e. edges whose source and target nodes are identical) are omitted. This can be done because as soon as we look at a Hasse diagram, we are supposed to know that there is an implicit reflexive edge for each node.
- Transitive edges are omitted, i.e., if there is an edge from {1} to {1,2} and one from {1,2} to {1,2,3}, there is no need to draw an additional transitive edge from {1} to {1,2,3}.
- By convention, all edges are drawn "upwards", so we don't have to paint arrows at their ends.

Figure 1 illustrates the Hasse diagram for the poset we already encountered a number of times.

6 Bounds

An **upper bound** u of two elements a and b in a poset is defined in the following way: Both $a \sqsubseteq u$ and $b \sqsubseteq u$ must be true. For example, the elements {1,2} and {1,3} have the upper bound {1,2,3}, which can be easily read from the Hasse diagram in Figure 1. The elements {1} and {1,2} have the following upper bounds: {1,2}, {1,2,3}. This shows that there can be multiple upper bounds, and that upper bounds don't have to be different from the elements they are computed for.

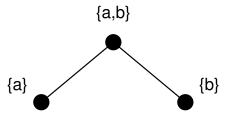

Figure 2: A poset that is not a lattice.

The **least upper bound** (also called **supremum** or **join**) is quite straightforward once you know what an upper bound is. It is simply an upper bound that is $\sqsubseteq$ all other upper bounds. For example, the least upper bound of {1} and {1,2} (written as $\{1\} \sqcup \{1,2\}$) is {1,2}, and $\{1,2\} \sqcup \{1,3\} = \{1,2,3\}$. The least upper bound is always unique[1], but doesn't have to be different from the elements it is computed for. The notions of **lower bounds** and **greatest lower bounds** (also known as **infimum** or **meet**) are defined analogously.

Upper and lower bounds (as well as their "least" and "greatest" variants) can also be computed for multiple elements (i.e. sets of elements) instead of only for two elements. For example, the least upper bound of {1}, {2}, and {3} is {1,2,3}. In short: $\bigsqcup(\{1\},\{2\},\{3\}) = \{1,2,3\}$

The **greatest element** of a set of elements inside a poset has the following property: It must be among the given set of elements, and all other elements in the given set must be $\sqsubseteq$ this greatest element. For example, the greatest element of the set {{1}, {1,2}, {1,2,3}} is {1,2,3}. The **least element** can be defined analogously to the greatest element.

Note that there also exist **maximal** and **minimal** elements, which are different from greatest and least elements. However, these notions are not necessary for the understanding of lattices and are skipped in this tutorial.

7 Lattices

A **lattice** is a poset in which *all nonempty finite* subsets have both a least upper bound and a greatest lower bound. We will come to speak about infinite subsets later, so don't get confused by this now. The powerset example from previous sections is such a lattice: No matter which and how many elements you take, you can always compute a least upper bound as well as a greatest lower bound for them. For a better understanding of the concept of lattices, it might be useful to see an example for a poset that is not a lattice, so one is given in Figure 2. It shows a poset with three elements ordered by subset inclusion. Since there is no greatest lower bound for the elements a and b, this poset doesn't satisfy the lattice condition.

A **complete lattice** is a poset in which *all* subsets have both a least upper bound and a greatest lower bound. Note the difference between complete lattices and "normal" lattices: While for complete lattices, the bounds condition has to be satisfied by all subsets, normal lattices have to satisfy it only for nonempty finite subsets. It can be shown that all lattices with a finite number of elements are always complete lattices. Hence, we will need a lattice with an infinite number of elements to demonstrate the difference. For instance, con-

[1] if it exists

sider a poset with all integers ($\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}$) as its elements and the usual $\leq$ as partial order relation. Obviously, this poset is infinite, and it satisifies the normal lattice condition. However, one of the subsets in this lattice is the set $\mathbb{Z}$ itself. Since neither a least upper bound nor a greatest lower bound can be computed for this subset, the condition for a complete lattice is not satisfied. We can simply turn this normal lattice into a complete lattice by adding the elements ∞ and $-\infty$ to the underlying set. Intuitively, this means that a complete lattice must always have a greatest element (the **top element**, written as $\top$) as well as a least element (the **bottom element**, written as $\bot$).

By the way: Note that the above definition of a complete lattice can be cut down to requiring only a least upper bound for all subsets, or only a greatest lower bound for all subsets. The reason is that one condition automatically implies the other, which can be shown mathematically.

Now we come to the illumination of a rather confusing aspect of complete lattices, which has to do with the **empty subset** (written as $\emptyset$). Beware: You must not confuse the empty subset of a lattice with a simple element of a lattice. For instance, consider a powerset lattice with the elements {{}, {1}, {2}, {1,2}}. One of the elements is the empty set {}, but this is *not* the empty subset $\emptyset$ of this lattice. You get the empty subset by taking *nothing* from the lattice, not even the element {}. So far, so good - here comes the tricky part. The following equation is true, even though it might appear odd at the first glance:

$$\bigsqcup \emptyset = \bot$$

$$\bigsqcap \emptyset = \top$$

Until now, we have already computed the least upper bounds for lattice subsets consisting of two or more lattice elements. The least upper bound of a single lattice element is quite straightforward, it is simply identical to the element itself. But what about the least upper bound of *no element*, as in this case? The key to this question is the fact that you can say virtually *everything* about *all elements of* $\emptyset$, simply because they don't exist. For instance, you can say "every element of $\emptyset$ is green", and it would be true because you won't find any element to refute this statement. Therefore, we can also say that every element of $\emptyset$ is $\sqsubseteq$ any element of the complete lattice L. This means that all elements of L are upper bounds of $\emptyset$. Hence, the *least* upper bound of $\emptyset$ is the *least* element of L, namely $\bot$. With an analogous argumentation, we get that the greatest lower bound of $\emptyset$ is $\top$. Note that $\emptyset$ is the only subset of L for which the greatest lower bound is *larger* than the least upper bound (that's what usually confuses people in the first place). Besides, all this also leads to the insight that complete lattices must always have one or more elements. Otherwise, the least upper bound of the empty subset would not exist - but according to the definition of a complete lattice, it has to exist. Normal lattices are happy because they don't have to think about issues related to $\emptyset$, since their definition only talks about "nonempty" subsets.

In older literature, you might find the term **semilattice**. As the name implies, it resembles a normal lattice, with the difference that only one of the two demands has to be satisfied: It is a poset in which all nonempty finite subsets have a least upper bound *or* a greatest lower bound. Compared to the normal lattice definition, the "and" was replaced by an "or". Depending on which of the two conditions a semilattice satisfies, it is called a **join-semilattice** or a **meet-semilattice**, respectively.

8 Conclusion

This tutorial should have provided you with a solid basic understanding of lattices and related concepts. In case you spotted any errors, omissions or imprecisions, please contact me under the email address given on the front page.