Quantum Integers

Andrei Khrennikov[a] and Moshe Klein[b] and Tal Mor[c]

[a] *International Center for Mathematical Modeling in Physics, Engineering, Economics, and Cognitive Science, Lineaus University, S-35195, Vaxjo, Sweden*
[b] *Gan Adam L.t.d p.box 5024 Tivone Israel 36087*
gan_adam@netvision.net.il
[c] *Computer Science Dept. Technion, Haifa 32000, Israel*

Abstract. In number theory, a partition of a positive integer *n* is a way of writing *n* as a sum of positive integers. The number of partitions of *n* is given by the partition function *p(n)*. Inspired by quantum information processing, we extend the concept of partitions in number theory as follows: for an integer *n*, we treat each partition as a basis state of a quantum system representing that number *n*, so that the Hilbert-space that corresponds to that integer *n* is of dimension *p(n)*; the "classical integer" *n* can thus be generalized into a (pure) quantum state $|\psi(n)>$ which is a superposition of the partitions of *n*, in the same way that a quantum bit (qubit) is a generalization of a classical bit. More generally, $\rho(n)$ is a density matrix in that same Hilbert-space (a probability distribution over pure states). Inspired by the notion of quantum numbers in quantum theory (such as in Bohr`s model of the atom), we then try to go beyond the partitions, by defining (via recursion) the notion of "sub-partitions" in number theory. Combining the two notions mentioned above, sub-partitions and quantum integers, we finally provide an alternative definition of the quantum integers [the pure-state $|\psi'(n)>$ and the mixed-state $\rho'(n)$], this time using the sub-partitions as the basis states instead of the partitions, for describing the quantum number that corresponds to the integer *n*.

Keywords: Integers, Partition, Superposition, Density Matrix, Recursion.
PACS: 03.65.Aa, 03.67.-a, 02.10.De

INTRODUCTION

In number theory, a partition of a positive integer *n* is a way of writing *n* as a sum of positive integers. At a first glance, it appears that there is no difference whether one writes *1+1=2* or *2=1+1*. But a closer observation shows that there may be a meaningful intuitive difference between these expressions: The first is a simple addition exercise between two numbers, while the latter describes a possible *representation* of the integer *2* as a sum of two (smaller) integers. The Trivial Representation *2=2* serves as another representation for the number *2*. In number theory, a partition is only made up of the numbers being summed and not from the order of the summation. For instance, all of the partitions of the integer *5* are:

5=5
5=4+1
5=3+2

5=3+1+1
5=2+2+1
5=2+1+1+1
5=1+1+1+1+1

Two sums that differ only in the order of their summands are considered to be the same partition (if the order also matters, the resulting set will not be called a partition anymore). A summand in a partition is also called a part and the number of partitions of n is given by the partition function p(*n*).

As a few other simple examples, *3*, *2+1*, and *1+1+1* are the three partitions of the integer *3*, and *4*, *3+1*, *2+2*, *2+1+1*, and *1+1+1+1* are the partitions of *4*. Note that p(*1)=1*, p(*2)=2*, p(*3)=3*, p(*4)=5*, and p(*5)=7*. It is interesting to mention that there aren't any explicit formulas yet that calculate the Partition Function for a given *n*.

To keep tracks of all partitions, we follow the method used in Wikipedia's entry "Partition (number theory)", and we write the larger number at the left-hand-side, so *3+1* appears here, while *1+3* (which corresponds to the same partition) does not appear. We also order the partitions by writing the partition having the larger number prior to the one having the smaller number, so that the partition *3+1* appears before the partition *2+2*, and *2+2* appears before *2+1+1*.

The different partitions of a number *n* describe something that resembles indistinguishability of items, e.g., if we talk about three similar items we might prefer to use the partition *3*, while if we talk about two similar items and one which is different from them, we might prefer to use the partition *2+1*. It is important to mention that the concept of partition has various uses in mathematics and physics, e.g., in group theory.

Here we extend the notion of partitions in two very different directions, and then we also combine the two directions. First, inspired by quantum information processing, we extend the concept of partitions in number theory as follows: for an integer *n*, we treat each partition as a basis state of a quantum system representing the number *n*, so that the Hilbert-space that corresponds to that integer n is of dimension *p(n)*; the "classical integer" *n* can thus be generalized into a (pure) quantum state which is a superposition of the partitions of *n*. This resembles the way a qubit is defined to be a generalization of a bit. More generally (and this is also true for the qubit of course), one can define a mixture of such quantum pure states, which is a density matrix in that same Hilbert-space.

Second, inspired by the notion of quantum numbers in quantum theory, such as *n*, *l*, and *m* that describe energy, angular momentum and the z-direction angular momentum in Bohr`s model of the atom, we try to go beyond the partitions. We define the notion of "sub-partitions"; this is done via using a recursion, and by the removal of redundant terms.

Finally, combining the two notions mentioned above, sub-partitions and quantum integers, we finally provide a second definition for the quantum integers, states (pure or mixed) that live in larger Hilbert-spaces, defined using the sub-partitions as the basis states instead of the partitions.

QUANTUM INTEGERS

We define the set that contains the partitions of n to be Par*(n)*. For instance, Par(*5*) is the "partition set of the number *5*". We call each of these partitions in the set, an element of Par*(5*). E.g. *{3+2}* and *{4+1}* are elements of Par*(5)*. For every positive integer n, the Partition Function, p(*n*), which gives the number of different ways to partition a number, is the number of elements in Par*(n)*.

One can define a probability distribution over the partition. E.g. for Par*(5)* we may define Prob*({5}*)*=p* and Prob*({4+1}*)*=1-p*. This probability distribution can be viewed as a (classical) generalization of the number 5. Inspired by quantum mechanics, we define a quantum variant of the natural numbers: For this, we use the bracket notation, and we let each partition be a basis state. E.g., the basis states for the number 5 are $|\varphi_1>=|5>$, $|\varphi_2>=|4+1>$, …, $|\varphi_7>=|1+1+1+1+1>$.

The above classical probability distribution can then be written as having the state $|\varphi_1>=|5>$ with probability p and having the state $|\varphi_2>=|4+1>$ with probability 1-p. Using these basis states, one can now easily go beyond the classical probability distribution, and define (in analogy to the definition of a qubit) pure quantum states to be superpositions of these basis states, and mixed quantum states to be mixtures of such pure states. As an example, $|\psi(5)>=\cos(\phi)|5>+\sin(\phi)|4+1>$ is an example of a pure state, namely a superposition of orthogonal basis states. Finally, here is an example of a mixed quantum state (a probability distribution over quantum states): $\rho(5)=q|\psi><\psi|+(1-q)|1+1+1+1+1><1+1+1+1+1|$, with $|\psi>$ being a superposition of the seven basis states written above.

Remark: An important feature of the quantum numbers model is a possibility to realize number-theoretic representations of dimensions different from 2^n. We remark that such spaces arise naturally in some models of quantum computing. As an example we can mention realization of clusters of entangled ions via interaction with laser pulses, see [1]. This is a consequence of appearance of so called dark levels. It might be that the construction presented in this paper would be useful for the mathematical representation of such models or others.

SUB-PARTITIONS

When we take a specific partition of a number *n* we may ask define sub-partitions. Say, if *n=k+m* is a partition of *n*, we can now consider a sub-partition using the fact that (for instance) *k=t+s*, namely using the partition of each of the summands. We here assume that this is different from simply taking the “other” partition *n=t+s+m* (after a proper ordering of the elements), since we want the sub-partition to be an internal process, defined after we already “finished” the partition (as if we are now sorting according to a different property). This resembles the way we write quantum numbers in Bohr`s model of the atom: we first divide according to the energy, and then for each energy we perform a further division according to the angular momentum, etc.

As our first example, we can see that the two partitions of the number *2*, namely *{2}* and *{1+1}* (where we use the set notations as in the previous section), cannot yield anything new if we try to perform a sub-partition, as the two counted items can either be identical (which corresponds to the partition *{2}*) or different (which corresponds to the partition *{1+1}*). There is nothing more to do in this case. For the number *2*, there are two partitions, and furthermore, there are also just two sub-partitions, and these are identical to its partitions. Looking at this differently: any attempt to have additional sub-partitions (further partitioning of *{1+1}* or of *{2}*) will lead via recursion to an infinite number of such sub-partitions.

As a more interesting example, if *{3}, {2+1},* and {*1+1+1},* are the partitions of *3*, we believe that now it does make sense to look at second-level partitions. As before, it is clear that it is meaningless to perform sub-partitions on the partition *(1+1+1)*, and on the partition *{3},* as this would lead again to an infinite number of sub-paritions via recursion. Thus, using the number *3* to help us finding a proper definition, we see that recursions may be used, yet must be applied carefully. For instance, in that case, it only makes sense to perform a sub-partition on the element *{2+1}* only. The number *2* has two different partitions "partition-a" which is *{2}* and "partition-b" which is *{1+1}.* The process of performing a sub-partition on the number *3* by using a partition of the number *2*, will thus lead to splitting the partition *{2+1}* into two sub-partitions: if we replace the summand *2* in the element *{2+1}* by its "partition-a" we get *{{2}+1}* and if we replace the summand *2* in the element *{2+1}* by its "partition -b" we get *{{1+1}+1}.* As this results from a sub-partitioning of the original partition *{2+1}* we consider the element *{{1+1}+1}* to be different from the element *{1+1+1}*, for the reasoning explained above (inspired by Bohr`s model): the first partition corresponds to one property (say, the energy) and the additional partitioning corresponds to an internal property (say, the angular momentum).

For the number *3*, we noticed that the elements *{3}* and *{1+1+1},* obtained at the first level of partition, cannot be further partitioned, and it only makes sense to consider further partitioning of *{2+1}.* After the second level of partition, we thus obtain four sub-partitions for the number *3*:

{3}
{{2} +1}
{{1+1} +1}
{1+1+1+1}

Note that after the first partition we had *{3}, {2+1}* and *{1+1+1},* therefore the notation we use here helps us to distinguish a partition such as *{2+1}* from a sub-partition such as *{{2}+1}:* note the set sign on the number 2, telling us that the process of sub-partitioning ended.

In the general case, for a partition of an integer *n*, in the first level of recursion, each element different from 1 and different from *n*, is replaced by its partition: Only the partition *{n}* and the partition *{1+1…+1}* do not go through a sub-partitioning, yet all other terms do. The partition *{(n-1) +1}* for instance, is replaced by *{{} +1}* in this first level of recursion, where each partition of *n*-1 (inserted in the internal parenthesis *{})* yields a different sub-partition of the type *{{} +1}.* One of these sub-partitions is *{{n-1}+1}*, that will not go through another sub-partitioning in the next level of recursion (as is true for any term *k* directly inserted in these *{}* parenthesis), yet terms like *{{(n-2)+1}+1}* or *{{(n-3)+2}+1}* or *{{(n-3)+1+1}+1}* will go through sub-partitioning in the next level of recursion (this is relevant for n larger than *4*). In general, terms that looked like $\{a_1+a_2+\ldots+a_k\}$ in the original partition are now (after that sub-partitioning) containing at least two summands, where each summand a_i is either the number 1, or it is replaced by { } where any partition of a_i can be inserted in these { } parentheses. For the number *3*, there is no need for a second recursion, but in general there can be many levels of recursion. At the second level of recursion the same process applies, and the deepest level of recursion ends only when any term *k* appears directly inside these parenthesis { }. This is why *{2+1}* did not describe a sub-partition at the end of the process, while *{{2} +1}* describes such a sub-partition.

Let us now look at the number *4*. The standard process leads to the partitions *{4}, {3+1}, {2+2}, {2+1+1},* and *{1+1+1+1}*. For the reasons explained above, we only need to perform a sub-partitioning on the summands *2* and *3*. From the regular partition of the number *2* in the partition *{2+1+1}* we get two sub-partitions *{{2}+1+1},* and *{{1+1}+1+1}*. From the partition *{3+1}* we get the following sub-partitions using the regular partition of the number *3: {{3}+1}, {{2+1}+1}, {{1+1+1}+1},* then using an additional level of recursion (this time on the summand *2* that appears in the term *{{2+1}+1})* we finally obtain: *{{3}+1}, {{2}+1}+1}, {{{1+1}+1}+1}, {{1+1+1}+1}*. Note that the same result can be obtained directly using all sub-partitions of *3*, instead of using its partition plus a second level of recursion.

We see that terms that look like *{k+1}* have a recursion over the number *k*. Obviously, terms that look like {*m*+*k*} have recursions over both m and k. However, we'll now see that there are some redundant elements in this case, if *m*=*k*: When we take the partition *{2+2}* and perform sub-partitioning of it we get *{{2} {2}}, {{1+1} {2}}, {{2} {1+1}}*, and *{{1+1} {1+1}}*. The second and third elements are identical once we ignore the order, thus one of them must be removed. We are finally left with the following *11* terms for the sub-partitions of *4*:
{4}, then *{{3}+1}, {{2}+1}+1}, {{{1+1}+1}+1},* and *{{1+1+1}+1},* then *{{2}{2}}, {{2}{1+1}},* and *{{1+1}{1+1}},* then *{{2}+1+1}*, and *{{1+1}+1+1},* and finally *{1+1+1+1}*.

Note that each step in the recursion corresponds to adding internal parenthesis. E.g., *{3+1}* is replaced by *{{} +1}* where we then input into *{}* all possible sub-partitions of *3*. Similarly, when we calculate the sub-partitions of the number *5*, we can make use of our earlier knowledge regarding all sub-partitions of *2, 3*, and *4*. This will lead to replacing each partition of *5* (except the trivial ones {*5*} and *{1+1+1+1+1})* by the relevant sub-partitions, so that *{4+1}* is replaced by *11* elements, *{3+2}* is replaced by *8* elements, etc. For the number *5,* the removal of redundancy also exists, as the partition *{2+2+1}* gives the sub-partitions *{{2}+{1+1}+1}* and *{{1+1}+{2}+1}* which are identical, and we keep only the first one.

Let us show now explicitly that there are *30* different sub-partitions of *5*: The partitions *{5}* and *{1 + 1 + 1 + 1 + 1}* do not have sub-partitions. The partitions *{4+1}, {3+1+1},* and *{2+1+1+1}* contribute *11, 4*, and *2* sub-partitions respectively, and the partition *{3+2}* contributes *2* times *4*, namely *8* sub-partitions. Finally, the partition *{2+2+1}* contributes 4 terms, but one of them is removed due to redundancy, leading to a total of *3* here, and a total of *30* altogether.

TABLE 1. Sub-partitions of small integers

The number N	Partitions on N	Number of sub-partitions
1	1	1
2	2	2
3	3	4
4	5	11
5	7	30
6	11	96

When we calculate the sub-partitions of the number *6*, we need to remove redundancy that will result from the terms *{3+3}, {2+2+2},* and *{2+2+1+1}*. The total number of partitions and sub-partitions for $n=1...6$ is given in Table 1.

The generalization now from these classical sub-partitions to a quantum integer based on the resulting Hilbert space is straight forward: We only need to replace the basis states that were defined in Section 2 (the partitions) by the sub-partitions defined here. E.g., the quantum integer *3* lives in a four-dimensional Hilbert-space in this case, instead of a three-dimensional one, and the quantum integer *4* lives in *11*-dimensional Hilbert-space instead of a five-dimensional one.

Remark: The quantum representation on the number theoretic construction of this paper is a natural generalization of the quantum representation based on expansions of natural numbers $x \in \{0,...,2^n - 1\}$ with respect to 2-adic scale:

$$x = x_0 + x_1 2 + ... + x_{n-1} 2^{n-1}, \;\; x_j = 0,1 .$$

This standard representation provides the enumeration of the basic vectors $|x_0 x_1 ... x_{n-1}\rangle$ in the n-qubit space by natural numbers: $x \equiv |x_0 x_1 ... x_{n-1}\rangle$. The main distinguishing feature of the representation constructed in this paper is the possibility to proceed with sub-partitions which is impossible to do in the standard n-qubit representation.

SUMMARY

We generalized the notion of integers into quantum integers. The motivation for that is currently purely abstract, yet we believe that this is an interesting concept that might have important impact in the future. One should also give some thought to the meaning of the classical probability distribution that corresponds to an integer. In particular, one may wonder what happens when such a number (the classical one or the quantum one) is observed, namely measured, to yield a specific partition (or sub-partition). One may also wonder about the non-classical bases, e.g., the Hadamard basis in case of the quantum number *2*.

In general, combining notions from number theory with concepts from quantum theory and quantum information processing might be found beneficial to either field, although we cannot see a specific application yet.

We also performed a non-trivial step beyond the well-known concept of “partition”, by defining sub-partitions using the process of recursion (while removing the redundant elements). To the best of our knowledge this is also a novel concept in number theory. Yet, it is important to mention here that a recent paper by Shadmi and

Klein [2] (also presented in [3]) already discussed a notion quite similar to our sub-partitions (they call their method "Organic Numbers"), which is also based on recursion and removal of redundancy, but their definition is not identical to ours.

Finally, we combined the two ideas to define an even more general concept of quantum integers, yet as before, we cannot see a specific application for that generalization yet.

ACKNOWLEDGMENTS

MK thanks Chen Kupperman who translated the first version of the paper from Hebrew and helped with his mathematical insight. MK also thanks Avi Berger for his important remarks. MK also thanks Aloni Cohen from SHAMRAZ Technological Services ltd. for the Sponsorship. AK and MK would like to thank the Profile Mathematical Modeling and System Collaboration of Vaxjo University for support for participation in the conference QTRF-5.

REFERENCES

1. Basieva Irina et al., "Quantum control of exiton states in clusters of resonantly interacting fluorescent particles using biharmonic laser pulses". Phys. Rev. B, volume **74,** 2006, 165329-165341.
2. Klein Moshe and Shadmi Doron , "*Organic Mathematics*", International Journal of Pure and Applied Mathematics, volume **49** No. 3 2008, 329-340
3. Klein Moshe, Shadmi Doron "*Organic Mathematics*", presentation at the 5 th' International Conference on Quantum Theory Reconsideration of foundation Vaxjo University Sweden 2009.