Finding Defectives on a Line by Random Docking and Interval Group Tests

Peter Damaschke

Department of Computer Science and Engineering, Chalmers University
41296 Göteborg, Sweden
ptr@chalmers.se

Suppose that some of the n elements of a totally ordered structure is defective, and several repair robots are at our disposal. They can dock at a random element, move at unit speed or leave, and send each other signals if there is no defective between them. We show that, by using only two robots that obey simple rules, the defective can be localized in $O(\sqrt{n})$ time, which is also optimal. A variation of our strategy needs three robots but has a more predictable behaviour. The model is motivated by a conjectured DNA repair mechanism, and it combines group testing with geometric search.

Keywords: Cooperative robots; interval group testing; cow path problem.

Mathematics subject classification: 68Q25, 68W15 97M60

1. Introduction

The following model of detecting and repairing defectives in a totally ordered set by robots that can move and communicate in simple ways is abstracted from a biological mechanism that is conjectured to play a major role in repairing DNA damages by specialized proteins [7].

Let $\mathcal{L}$ be a totally ordered set of n elements. We think of $\mathcal{L}$ as a discretized horizontal line segment, such that we can use the terms "left" and "right". At times, a few elements may become defective. There is also a fleet of robots each of which is capable of, in short, the following actions:

- place oneself on a random element of $\mathcal{L}$, in one time unit
- move to the next element to the left or to the right, in one time unit
- send a signal along $\mathcal{L}$, in one direction or in both directions
- receive signals sent by other robots on $\mathcal{L}$
- repair a defective element if being there
- change one's internal state depending on other events (see below)
- leave $\mathcal{L}$

Furthermore, we assume that *a defective or a robot stops any incoming signal*, whereas an intact element without a robot on it passes the signal to the next element. To motivate this special assumption, some details of the biological background are needed. Our set $\mathcal{L}$ models a DNA string, and the robots model certain repair proteins. They can send signals along the string, but any defective (a physical damage, or bases on the complementary strings that do not form a proper base pair A-T or C-G) interrupts the string, such that the signal does not go through, and similarly, a repair protein that has docked at the string causes an interruption, too. As we discuss below, this interruption of signals allows a fleet of "robots" to detect defectives efficiently by some group testing strategy.

The question is how fast the robots can detect and repair defectives, by appropriate local rules. These rules determine when to send signals and how to move, and actions may depend on the received signals, because these are the only sources of information for the robots).

Before we tackle the problem, we comment on the above assumptions to clarify the model.

The robots are mobile automata that have internal states. They can sense events like a collision with another robot, an incoming signal, its content, and the direction it comes from. According to some built-in deterministic or randomized rule the robot will then switch to another state.

We imagine that robots not being on $\mathcal{L}$ move around blindly in free space (in the biological scenario: in the cell) and then dock at a random element of $\mathcal{L}$. No element is preferred, thus we make the simplest possible assumption of a uniform distribution. Moreover, all choices of elements are independent; this applies both to the choices made the same robot over time, and to choices made by different robots.

Without loss of generality we normalize the units of measurement as follows. To the define the time unit, we assume that the free movement outside $\mathcal{L}$ always takes one time unit. That is, a robot that has just left $\mathcal{L}$ docks again at a random element after one time unit. (Alternatively, it would also be natural to assume an exponential distribution of the duration until the next docking, with the expectation as time unit, but this would be more complicated and would yield essentially the same results as we will present.) Robots move with fixed speed at $\mathcal{L}$. Next we normalize the length unit such that this fixed speed becomes unit speed. Furthermore, instead of a continuous line subdivided into unit-length segments we think of a discrete ordered set, which is only a notational difference. Several robots may occupy the same element of $\mathcal{L}$, and in this case their ordering is arbitrary.

We do not assume that the robots act in discretized and synchronized time slots (as this would be inappropriate, given the motivation). In particular, if several robots arrive at $\mathcal{L}$ within one time unit, their exact arrival times are still pairwise distinct.

Unlike the robots that move at unit speed, signals are propagated instantly, that is, whenever a robot emits a signal, this is immediately noticed by the other robots that are on the side of the signal and are not separated from the sender by a defective or by another robot. (More precisely, the signals' speed is so large compared to the robots' speed that we can assume infinite speed.)

Robots are not assumed to have a global sense of direction, that is, they have no common concept of left and right, but they can locally distinguish the two directions and notice the direction a signal came from.

2. Our Result and its Relation to Group Testing

According to the model, a robot can test the element at its current location for being defective, and two robots can test by signalling whether there is a defective between them. In other words, robots can do certain group tests. In the original combinatorial group testing problem (see [6] for a general introduction), an unknown small subset of a given set of n elements consists of defective elements, and a searcher can choose any subsets and test it for the presence of defectives. The answer is binary, and in the positive case neither the number nor the identities of defectives are revealed. Thus, in the negative case one can remove the tested subset, and in the positive case, further group tests must narrow down the candidate set. Actually, a set of d defectives can be identified by $O(d \log n)$ tests, straightforwardly by bisection search. (We refer again to [6] for basic facts.)

In interval group testing, the set is totally ordered, and group tests are restricted to consecutive subsets. Still d defectives can be identified by $O(d \log n)$ tests [3,4,5]. Motivated by another DNA related matter, interval group testing with a further restriction is studied in the mentioned papers: Only a few rounds of simultaneous tests are permitted. For two rounds, the number of tests goes as $O(\sqrt{n})$ for any fixed d.

What distinguishes the problem considered in the present paper from virtually all earlier literature about group testing variants is that no powerful searcher can freely decide which subsets shall be tested next and is therefore, in principle, able to choose the most informative tests. Instead, the ends of intervals tested by two robots are partly random and beyond the robots' control, and single elements can be tested only in the given order, starting from the currect location of a robot. Therefore, usual group testing strategies do not apply here.

Our actual problem is to design local rules for the robots that make up a strategy to efficiently detect and repair defectives. Our contributions can be summarized as follows: For the problem described in Section 1 we propose in Section 3 a search strategy for robots with the mentioned very restricted abilities. After some mathemetical preparations in Section 4, we show in Section 5 that any defective can be found in $O(\sqrt{n})$ expected steps, even by only two robots. After this main result we informally discuss in Section 6 an alternative $O(\sqrt{n})$ time strategy that requires three robots but yields a smaller variance of the search time. In Section 7 we propose further research directions.

The reader should not be misguided by the word "robot" which is only meant as a synonym for mobile devices that can move and communicate in specific ways, follow rules, and perform some simple automatic actions according to a protocol. They should not be confused with macroscopic mechanical robots, and it is not the aim of the paper to contribute to the field of robotics in that sense. (Likewise we could use one of the terms agent, automaton, device, machine, etc.) Besides "robotic" mechanisms in nature, recent technology is able to produce artificial molecular machines (as, e.g., the work of the Nobel prize laureates in chemistry 2016) each of which has only simple functionality. The present work aims to give one example of a study of the complexity of strategies that solve a cooperative task performed by machines of this kind. The chosen example comes from a real biological task and plausible assumptions about it. On the other hand, we do not claim that the presented strategy accurate models the real DNA repair mechanism. Rather, this theoretical study explores which simple abilities and rules would, in principle, enable which search times.

There is a vast amount of literature about other geometric search problems involving moving robots. The classic one-dimensional example is the cow path problem where a robot has to find an object on a line, knowing neither the direction nor the distance to the object. The robot detects the object only if it is at the same location. The meaningful question is then to minimize the competitive ratio, that is, the ratio of the travelled distance and the original distance to the object. (Assuming a robot that moves at unit speed, we can, equivalently, measure time rather than distance.) A certain zigzag strategy has the optimal competitive ratio 9, which can be improved to 4.5911 by randomization [2,8]. A group testing problem with a moving defective, rather than moving search robots, is studied in [1] on lines, cycles, and trees.

3. A Search Strategy

We consider a monitoring process that runs forever. A fleet of robots are patrolling. As long as $\mathcal{L}$ is free of defectives, the robots only examine $\mathcal{L}$. Once in a while a single defective comes up at some element, and the robots are to notice the presence of a defective and then localize it as soon as possible.

For this purpose we devise a strategy specified by the rules below. Let R and S be variables for robots, where R denotes a robot that has just arrived at a random point on $\mathcal{L}$, and S denotes any robot that is already on $\mathcal{L}$. Every newly arrived robot "switches from R to S" after one time unit, and it stores this role as a part of its internal state. That is, every robot always "knows whether it is an R or an S". The robots will send different signals encoding different events. This does not require additional assumptions, as a "signal i" may simply consist of i signals sent shortly one after another. The protocol describes an anytime monitoring process, therefore no specific initial state is defined, or equivalently, the initial state is arbitrary.

The strategy consists of a list of rules below. Apparently the list cannot be condensed, however, for an easier understanding we first describe the ideas informally. A defective partitions $\mathcal{L}$ in two "sides", the parts of $\mathcal{L}$ to the left and to the right of the defective. After docking, any robot (R) explores its neighborhood by sending here-I-am signals (1) in both directions and awaiting answers. Any robot (S) getting the signal from the direction it is moving to (towards a conjectured defective) learns that a new robot takes care of this defective, and there is no defective between the robots (as it would have interrupted communication), hence S leaves. Before that, it sends an answer (2) to the new robot R, informing it that it should search in the opposite direction. R becomes S and starts moving accordingly. But if S receives that signal from the direction it comes from, it informs the new robot by a signal (3) saying "I am already closer to the defective than you", urging the new robot to leave. Any new robot (R) docking at an "empty" side of the defective, where other robots are neither present at $\mathcal{L}$ nor arrive within a time unit, will not receive any response and will leave. However, with probability at least $1/2$, any robot being currently not at $\mathcal{L}$ will hit the longer side. In particular, the first robot arriving at the longer side after creation of the defective receives with constant probability a signal from a second robot, causing it to stay at $\mathcal{L}$ and to move in the other direction, where a defective may be waiting. The other robot moves as well. Now one of them approaches the defective, and one approaches the end of $\mathcal{L}$ and simply leaves when it is found.

Now we state the rules one by one:

- Upon arrival, R sends a signal 1 in both directions.
- If S receives a signal 1 from the direction it is moving to, then S sends a signal 2 back and leaves $\mathcal{L}$.
- If S receives a signal 1 from the direction it is moving away from, then S sends a signal 3 back and keeps on moving.
- If S receives a signal 1 while not moving, then it sends a signal 3 back, and starts moving in the other direction (where no signal came from).
- If R does not receive a signal after waiting one time unit, then R leaves $\mathcal{L}$.
- If R receives a signal 1 or 2, then R starts moving in the other direction (where no signal came from).
- If R receives a signal 3, then R leaves $\mathcal{L}$.
- If S reaches the defective, then S repairs it and leaves $\mathcal{L}$.
- If S reaches an end of $\mathcal{L}$, then S leaves $\mathcal{L}$.
- Any robot that has left $\mathcal{L}$ chooses a new random point on $\mathcal{L}$ after one time unit.

The following diagram illustrates a typical situation – the mechanism that leads to efficiency (as we will see). A new robot docks, luckily, at a position between the defective (d) and a robot moving to that defective. By the signalling they recognize that the new robot is closer to the defective and can take over, whereas the old robot leaves, to try a new random docking point. The $\Leftarrow$ arrows symbolize moving, and the simple arrows depict the signals exchanged.

d		$\Leftarrow S$
$d\ 1 \longleftarrow$	R	$\longrightarrow 1 \Leftarrow S$
d	R	$2 \longleftarrow\ \Leftarrow S$
d	$\Leftarrow R$	
d	$\Leftarrow S$	

4. Expected Time Until Success

In order to analyze the proposed strategy we first consider only the robots arriving on one side of a defective and moving towards the defective. The elements of $\mathcal{L}$ are indexed from left to right by integers, where the defective is at $x = 0$ and our robots arrive at positions $x > 0$. Let the defective show up at time $t = 0$.

Suppose for now that in each time unit $t \geq 1$ and at each element $x \geq 0$, with a fixed probability p some robot arrives, and these arrival events are independent. We wish to calculate the expected time until the defective is found.

For robots behaving as in the proposed strategy, the following lemma merely specifies which robot finds the defective at which time; this statement does not depend on p.

Lemma 4.1. *The time until the defective is found equals $y := \min_P(x + t)$, where P is the set of all pairs (x, t) such that some robot arrives at position x at time $t > 0$.*

Proof. Since robots can move by only one element per time unit, no robot can reach the defective strictly before time y. Below we show that y time units are also sufficient.

Consider the arrivals of robots in their termporal order. As soon as a robot moves to the left, any new robot arriving to the right of the moving robot receives signal 3 and leaves, whereas the moving robot does not change its direction. Any new robot arriving to the left of the moving robot receives signal 2 from the right and thus moves to the left as well. Thus, all moving robots will henceforth move to the left. It follows that the robot with minimum $x + t$ reaches the defective, at time y. □

In the following we use that the expectation of a random variable with positive integer values, $E[Y] = \sum_{k=1}^{\infty} k \cdot Pr(Y = k)$, can also be written as $E[Y] = \sum_{k=1}^{\infty} Pr(Y \geq k)$.

Proposition 4.2. *Consider the elements of $\mathcal{L}$ at one side of the defective, and let $p < 1$ be fixed. Suppose that, at every position x and in every time unit t, with probability p one robot arrives, and these arrival events for all pairs (x, t) are independent. Then some of these robots finds the target after an expected time $O(\sqrt{1/p})$.*

Proof. Let $q = 1 - p$. Let Y be the random variable that attains the value y as defined in Lemma 4.1. The time we wish to calculate is $E[Y]$.

Remember that $x \geq 0$ and $t \geq 1$. We have $Y \geq y$ if and only if no robot appears at any of the $\binom{y}{2}$ points (x, t) with $x + t < y$. Since arrivals are independent, the probability of this event is $q^{\binom{y}{2}}$, which yields:

$$E[Y] = 1 + \sum_{y=2}^{\infty} q^{\binom{y}{2}}.$$

By replacing $\binom{y}{2}$ with the smaller $(y-1)^2/2$ and shifting the index we obtain:

$$E[Y] \leq 1 + \sum_{y=1}^{\infty} q^{y^2/2}.$$

Since the summands are values of a monotone decreasing function in y, this further implies:

$$E[Y] \leq 1 + \int_0^{\infty} (1-p)^{y^2/2} dy.$$

Using $(1-p)^{1/p} < e^{-1}$ (inverse of Euler's number) for all $p < 1$, this is bounded by:

$$E[Y] \leq 1 + \int_0^{\infty} e^{-(p/2) \cdot y^2} dy.$$

A charming fact is that the integrand is, up to a scaling factor, the density function of the normal distribution, that is, this definite integral is well known, By variable substition it finally evaluates to:

$$E[Y] \leq 1 + \sqrt{2/p} \cdot \sqrt{\pi}/2 \leq 1 + 1.2534\sqrt{1/p} \qquad \square$$

Note that the hidden constant factor in $O(\sqrt{1/p})$ is small.

Now consider a scenario where n elements are on one side of the defective, and the robots that are currently not on $\mathcal{L}$ arrive at random elements of $\mathcal{L}$ in each time unit. Thus our arrival probability is $p = \Theta(1/n)$ for all $x \leq n$. Proposition 4.2 now suggests that any upcoming defective is found after $O(\sqrt{n})$ time. With a fixed number of robots, arrivals in the same time unit are no longer independent, but we only need to modify the argument slightly:

Proposition 4.3. *Suppose that n elements are on one side of the defective, and in every time unit one robot arrives at one element chosen uniformly at random, where the choices in different time units are independent. Then some of these robots finds the defective after an expected time $O(\sqrt{n})$.*

Proof. With $p = 1/n$ and $q = 1 - p$, the probability of $Y \geq y$, for $y \leq n$, is now:

$$Pr(Y \geq y) = \prod_{k=1}^{y-1} (1 - k/n) = \prod_{k=1}^{y-1} (1 - kp) < \prod_{k=1}^{y-1} (1-p)^k = q^{\binom{y}{2}}.$$

Since $Pr(Y \geq y) = 0$ for $y > n$, the same bound upper trivially holds there, too. From now on the calculation is the same as in Proposition 4.2. $\square$

5. Two Robots Against One Defective

After this preparation we turn to the actual search problem where one, unknown element of an ordered set $\mathcal{L}$ of n elements has just become defective, and the patrolling robots are to find it.

Theorem 5.1. *Two robots find a defective after an expected time $O(\sqrt{n})$.*

Proof. The rules of the strategy cause the following behaviour.

As long as no defective is present, one robot S is on $\mathcal{L}$, moving towards one end, the other robot R probes random elements of $\mathcal{L}$, and sometimes they switch their roles. At the moment when a defective arises, S is on one side of the defective, say on the right side. Assume that S is moving to the left at this moment.

Each time when R chooses a random element on the right side, one step of the process in Proposition 4.3 is done. Whenever R arrives on the left side, it leaves $\mathcal{L}$ after one time unit since its signals are not answered. Moreover, S does not notice anything and keeps on moving. Thus, the effect is the same is if R would arrive to the right of S. It follows that the time bound from Proposition 4.3 applies, subject to a (small) constant factor due to R's waiting times on the left side.

Hence, if S moves to the left as assumed above, some robot finds the defective after an expected time $O(\sqrt{n})$. However, if S moves to the right, then the right end of $\mathcal{L}$ is found instead of the defective. But then both robots are away from $\mathcal{L}$ for one time unit, and both robots begin choosing random elements independently. Remember that robots are not synchronized, and one robot arrives earlier, even within the same time unit. With constant probability they also arrive at the same side, in which case the process starts over. (Note that the probability of both robots arriving on the longer side is at least 1/4). In particular, the earlier, waiting robot moves to the left or right with probability 1/2. Before this event, the robots have repeatedly chosen elements on different sides and then left $\mathcal{L}$, but this happens only $O(1)$ times on expectation.

Altogether, after an epoch of $O(\sqrt{n})$ expected time units, the defective is found with probability 1/2. The strategy repeats such epochs independently until the defective is found. Thus, the expected overall time is the sum of the expected times of the first N epochs, where N itself is a random variable which is independent of the epoch lengths and has expectation $E[N] = 2$. Using Wald's equation, the total expected time is still $O(\sqrt{n})$. □

It is not hard to see that, under the introduced model, the $O(\sqrt{n})$ result in Theorem 5.1 is tight in several ways:

1. More than two robots following the proposed strategy would only improve the constant factor in $O(\sqrt{n})$, but not the $O(\sqrt{n})$ time bound itself.

2. Recall that, once a robot moves in the correct direction towards the defective, our strategy ensures that the robot closest to the defective stays on $\mathcal{L}$. This is already the best one can do in this search model (as we have no control over the arrival

points of robots which are random). Thus, no other strategy can be faster than $O(\sqrt{n})$ either.
3. A single robot can only move at unit speed, starting from its random docking point, and check the traversed parts of $\mathcal{L}$ for the presence of a defective, thus it needs $\Theta(n)$ expected time. In other words, at least two robots are necessary to achieve $O(\sqrt{n})$ expected time.

It is also enlightening to consider weaker models. Trivially, a fixed number of robots being unable to move on $\mathcal{L}$ would need $\Theta(n)$ expected time, since one of them must exactly hit the defective. A fixed number of robots that can move but not communicate would need $\Theta(n)$ expected time, for the same reason as in the one-robot case above.

6. Three Robots Against One Defective

In this complementary section we give a variation of our search strategy, with the same expected time of $O(\sqrt{n})$, but with somewhat different properties. Recall that R denotes a robot that has just arrived at a random point on $\mathcal{L}$, and S denotes any robot that is already on $\mathcal{L}$.

- Upon arrival, R sends a signal in both directions.
- If S receives a signal from the direction it is moving to, then S sends a signal back and leaves $\mathcal{L}$.
- If S receives a signal from the direction it is moving away from, then S sends a signal back and keeps on moving.
- If S receives a signal while not moving, then it sends a signal back, and starts moving in the other direction (where no signal came from).
- If R does not receive a signal after waiting one time unit, then R leaves $\mathcal{L}$.
- If R receives a signal from exactly one direction, then R starts moving in the other direction (where no signal came from).
- If R recevies a signal from both directions, then R leaves $\mathcal{L}$.
- If S reaches the defective, then S repairs it and leaves $\mathcal{L}$.
- If S reaches an end of $\mathcal{L}$, then S leaves $\mathcal{L}$.
- Any robot that has left $\mathcal{L}$ chooses a new random point on $\mathcal{L}$ after one time unit.

We discuss the modifications. First it is obvious that only one type of signals is used. The downside is that the strategy needs three rather than two robots. After each arrival of the third robot, the middle robot leaves, the inner robot approaches the defective, and the outer robot approaches the end. This, however, has the advantage that only one epoch (in the sense as in the proof of Theorem 5.1) is needed. The expected time is smaller by a constant; this is not too remarkable, since more robots are involved. However the main point is: For large n, the distribution of the actual search time is concentrated around the expected value, whereas the two-robot strategy runs in an exponentially distributed number of epochs, such that the search time has a high variance. In further work it might be interesting to analyze the variance for an increasing number of robots.

7. Conclusions and Further Research: Many Defectives

We studied a model of robots that find and repair defectives on a totally ordered set. The model is inspired by a possible repair mechanism in molecular biology but might also find applications in other domains, when generalized to networks. For instance, think of networks of pipelines or electrical conductors where moving robots can test whether the paths between them are intact, or a leak or interruption has appeared.

The model is a hybrid of interval group testing and geometric search problems like the cow path problem, with the special feature that part of the movement is random and cannot be controlled by a searcher. Our main insight is that random docking, deterministic moving, and very restricted signalling are sufficient to locate a defective in $O(\sqrt{n})$ expected time with a fixed number of robots, and this is optimal in a sense.

As opposed to usual "non-geometric" group testing problem variants where the case of one defective is often trivial, single defectives are the most interesting matter in our case, both from the motivation (one would not expect too many simultaneous damages) and from the algorithmic perspective: The analysis also applies to multiple defectives. The first defective approached by a moving robot is found after $O(\sqrt{n})$ expected time as before, and so are the others, one by one. A minor issue is that our strategy requires two robots arriving within one time unit between two defectives, because a single robot getting no response to its signals would leave. In the presence of d defectives it takes $O(d)$ expected time until two robots cooperate. Alternatively, in a slightly modified strategy, any new robot not receiving answers may start moving in a random direction rather than leaving. This modified strategy would be more efficient in the case of many defectives, but it would use unnecessarily many robots if defectives are rare. Further research can also address the long-term bahaviour: If defectives appear frequently, say at a fixed rate and at random positions, what can be proved regarding the power of a fleet of patrolling robots? How many robots are needed to prevent an accumulation of defectives? These questions seems to require methods from queuing theory and Markov chains. Moreover, one could consider an accompanying mechanism that would create or phase out robots, in order to adapt their number to a changing rate of defectives. (The amount of repair proteins in a cell may vary.) An intriguing question in such a setting is whether repairing defectives near the end needs more expected time than in the middle, because robots can approach them from only one side. Does this explain the gradual loss of end segments of chromosomes?

Acknowledgement

I would like to thank my colleague Graham Kemp for hinting me to the paper [7] and its connection to group testing. An early draft was presented at the First Aachen Discrete Meathematics Days in 2016, an informal workshop (without proceedings) on the occasion of Eberhard Triesch's 60th birthday. I thank the workshop organizers for the kind invitation and the participants for inspiring discussions.

References

[1] H.K. Aydinian, F. Cicalese, C. Deppe, V. Lebedev, A combinatorial model of two-aided search, in SOFSEM 2016, LNCS 9587, eds. R. Freivalds, G. Engels, B. Catania (Springer, 2016), pp. 149–160.
[2] R.A. Baeza-Yates, J.C. Culberson, G.J.E. Rawlins, Searching in the plane, Info. and Comp. 16 (1993) 234–252.
[3] F. Cicalese, J.A. Amgarten Quitzau, 2-Stage fault tolerant interval group testing, in ISAAC 2007, LNCS 4835, ed. T. Tokuyama (Springer, 2007) pp. 858–868.
[4] F. Cicalese, P. Damaschke, L. Tansini, S. Werth, Overlaps help: Improved bounds for group testing with interval queries, Discr. Appl. Math. 155 (2007) 288–299.
[5] F. Cicalese, P. Damaschke, U. Vaccaro, Optimal group testing algorithms with interval queries and their application to splice site detection, Int. J. Bioinf. Res. Appl. 1 (2005) 363–388.
[6] D.Z. Du, F.K. Hwang, Combinatorial Group Testing and Its Applications, Series on Appl. Math. vol. 3 (World Scientific, 2000).
[7] M.A. Grodick, N.B. Muren, J.K Barton, DNA charge transport within the cell, Biochemistry 54 (2015) 962–973.
[8] M.Y. Kao, J.H. Reif, S.R. Tate, Searching in an unknown environment: An optimal randomized algorithm for the cow-path problem, Info. and Comp. 131 (1996) 63–79.