which are powers of primes $\leq B$. Then Hasse's Theorem tells us that, if p is such that $p+1+2\sqrt{p} < C$ and the order of $E \bmod p$ is not divisible by any prime $> B$, then k is a multiple of this order and so $kP \bmod p = O \bmod p$.

Example 3. Suppose we choose $B = 20$, and we want to factor a 10–decimal–digit integer n which may be a product of two 5–digit primes (i.e., not be divisible by any prime of fewer than 5 digits). Then choose $C = 100700$ and $k = 2^{16} \cdot 3^{10} \cdot 5^7 \cdot 7^5 \cdot 11^4 \cdot 13^4 \cdot 17^4 \cdot 19^3$.

We now return to the description of the algorithm. Working modulo n, attempt to compute kP as follows. Use the repeated doubling method to compute $2P$, $2(2P)$, $2(4P), \ldots$, $2^{\alpha_2}P$, then $3(2^{\alpha_2})P$, $3(3 \cdot 2^{\alpha_2}P), \ldots$, $3^{\alpha_3}2^{\alpha_2}P$, and so on, until finally you have $\prod_{\ell \leq B} \ell^{\alpha_\ell}P$. (Multiply successively by the prime factors ℓ of k from smallest to largest.) In these computations, whenever you have to divide modulo n, you use the Euclidean algorithm to find the inverse modulo n. If at any stage the Euclidean algorithm fails to provide an inverse, then either you find a nontrivial divisor of n or you obtain n itself as the *g.c.d.* of n and the denominator. In the former case, the algorithm has been successfully completed. In the latter case, you must go back and choose another pair (E, P). If the Euclidean algorithm always provides an inverse — and so kP modulo n is actually calculated — then you must also go back and choose another pair (E, P). This completes the description of the algorithm.

Example 4. Let us use the family of elliptic curves $y^2 = x^3 + ax - a$, $a = 1, 2, \ldots$, each of which contains the point $P = (1, 1)$. Before using an a for a given n, we must verify that the discriminant $4a^3 + 27a^2$ is prime to n. Let us try to factor $n = 5429$ with $B = 3$ and $C = 92$. (In this example and the exercises below we illustrate the method using small values of n. Of course, in practice the method becomes valuable only for much, much larger n.) Here our choice of C is motivated by our desire to find a prime factor p which could be almost as large as $\sqrt{n} \approx 73$; for $p = 73$ the bound on the number of $\mathbf{F}_p$-points on an elliptic curve is $74 + 2\sqrt{73} < 92$. Using (2), we choose $k = 2^6 \cdot 3^4$. For each value of a, we successively multiply P by 2 six times and then by 3 four times, working modulo n, on the elliptic curve $y^2 = x^3 + ax - a$. When $a = 1$ we find that the multiplication proceeds smoothly, and it turns out that $3^4 2^6 P \bmod p$ is a finite point on $E \bmod p$ for all $p|n$. So we try $a = 2$. Then we find that when we try to compute $3^2 2^6 P$, we obtain a denominator whose g.c.d. with n is the proper factor 61. That is, the point $(1, 1)$ has order dividing $3^2 2^6$ on the curve $y^2 = x^3 + 2x - 2$ modulo 61. (See Exercise 5 below.) Thus, our second attempt succeeds. By the way, if we try $a = 3$ we find that the method gives the other prime factor 89 when we try to compute $3^4 2^6 P$. (Usually, but not always, the method gives the smallest prime factor.)

Running time. The central issue in estimating the running time is to compute, for a fixed p and a given choice of bound B (which is chosen in some optimal manner), the probability that a randomly chosen elliptic curve modulo p has order N not divisible by any prime $> B$. Now the