Deadtime Corrections

July 7, 94 T.J. Liu

One of the methods we tried came from David Kawall, which we will refer to as method 1. It tries to predict the true rate from the obsered P_0 (Zero trigger probability) , using Poisson distribution. For each run, we have tfreq(i), which is the number of spills when we get i hits on mainor, where i goes from 0 to 16. We also bin the charge of each spill into different q bins. We now have qx(i) and qy(i).

qx(i): charge value of ith bin.

qy(i): number of spills of ith bin.

assume for a given charge, for a given spectrometer and beam polarization,

number of counts=cq (c is a constant and q is the charge)

For one spill, $P_0 = e^{-cq}$. Then we sum up all the spills, we have:

$$P_0 = \frac{\sum qy(i) \times e^{-c \times qx(i)}}{\sum qy(i)}$$

On the other hand, from tfreq(i), we have:

$$P_0 = \frac{tfreq(0)}{\sum_{i=0}^{i=16} tfreq(i)}$$

In this way, we have an equation to solve for c. Then the true counts are given by

$$R_{real} = \sum c \times qy(i) \times qx(i)$$

while the observed counts are:

$$R_{ob} = \sum_{i=1}^{i=16} min(4, i) \times tfreq(i)$$

then the correction factor is given by;

$$deadfac = \frac{R_{real}}{R_{ob}}$$

Calculated the dead time correction factors using this method, and found that some runs have resonable correction factors, say, for an average rate of 1, the factor is 1.02. but some others do not make sense, like for an average rate of .5, some give a correction factor of say, 1.2. And these "good" and "bad" runs appear in groups. Then I used a Monte Carlo program, assuming a dead time of 32ns, using the true rate derived by the above method, to reproduce tfreq distribution for each run, and saw that for the "good" runs, the reproduced tfreq agrees quite well with the observed one, but for the "bad" runs, they are way off. Tried to vary the dead time all the way up to 400 ns for these "bad" runs, still did not see agreement. Then noticed that the transition between "good" and "bad" runs happen when we are switching from electron runs to positron ones, or when we are changing the prescalar settings. Then I realized that because of the prescalars, even without dead time and with one q bin, tfreq would not be Poisson. So trying to predict the true rate using Poisson distribution is incorrect. (See appendix for more detail). In our system, the main trigger coming into mainor is not prescaled while the Pion trigger is, that makes it more complicated. However, when we are doing normal electron runs and prescalar values are set high, the mainor is dominated by the main trigger, so method 1 gives reasonable factors. When the prescalar values are set low, or when we are doing positron runs, prescaled Pion trigger is no longer negligible, that is why we get crazy numbers.

So we are forced to go back to the old E143 method, which we will call "method 2". This method is what we are using now. Assuming a spill time and a dead time, to generate the probability matrix M(i,j), which is the probability of observing i hits when there were really j hits, using Monte Carlo. We have:

$$tfreq(i) = \sum_j M(i,j) \times rtfreq(j)$$

where tfreq (rtfreq) is the observed(real) trigger frequency distribution. In principal, j in the sum goes to infinity, but because we are getting practically no hits beyond 10, it is safe to take j only up to 16. Then we invert this matrix M, to solve for rtfreq from tfreq, and define our correction factor as

$$deadfac = \frac{\sum_{i=1}^{i=16} i \times rtfreq(i)}{\sum_{i=1}^{i=16} min(4,i) \times tfreq(i)}$$

Looked at mainor time distribution in each run, it represents the spill shape. All the runs I looked at are around 2200 ns width. Also looked at adjacent mainor timing, there is a sudden rise at 32 ns. So use 2200 ns as beam time, and 32 ns as the dead time. The correction factor vs. rate looks quite smooth. Then plotted the correction factor vs. rate for the "good" runs only, using method 1, from sp4, compared with what we get from method 2, using 2200 ns as beam time. They agree quite well. (because there are a fairly large percentage of Pion trigger rate in mainor for sp7, so did not use method 1 for sp7 as comparison). Then plotted the correction factor vs. rate for sp4 using method 2, for a beam time of 1800 ns and 2600 ns. Compare these with the "good" runs plot using method 1 mentioned above, we see 1800 ns gives too high a correction, while 2600 ns gives too low a correction.

So, the method we decide to use is method 2, with a beam time of 2200 ns, and a dead time of 32 ns. We use the same method, but with beam time of 1800 ns and 2600 ns as the upper and lower limit, to get the error. We see that the correction factor itself is accurate to a few parts in 1000, and the error for the asymmetry is less than 2E-5.

To check the accuracy of the probability matrix M, used the Monte Carlo program but different trial number, the Change in M is negligible. Also checked some element of M by hand, they agree.

Appendix

Assume there is no deadtime, and just one q bin, then tfreq seems to be given by poisson distribution. But because Pion trigger is prescaled, that makes tfreq distribution different from possion distribution. A handwaving proof is that, Possion distribution is generated when the probablity of registering one hit in a small time interval dt is proportional to dt. But with prescalars, the moment after the prescalar is cleared, there is NO probability of registering one hit, and the probability increases with time,until the next clearance. So it is no longer uniform in time. Following gives more detail:

Assume for now that only Pion trigger comes into the mainor and is prescaled by a factor of N, and the observed average rate is :

$$orate = \frac{\sum i \times \texttt{tfreq(i)}}{\sum_{i=1}^{10} t_{\text{freq}}(i)}$$

so the true average rate before the prescalar must be: trate=orate*N , and ppois(i) (the probability of finding i hits in one spill BEFORE the prescalar) is Possion distribution. Now let's try to produce tfreq(i)(which is AFTER the prescalar) from ppois(i). Let's assume N=2 to make it easier to understand. Taking one spill during the run, at the begging of it, the prescalar might have received 0 or 1 hit. Because the total hits during the run is much larger than the prescalar value 2, each case has the same probability 1/2 . To find out tfreq(0), let's say When in case 1(prescalar has received 0 hit), if this spill had 0 or 1 hit, then the prescalar still woudn't fire. This probability is $1/2*(ppois(0)+ppois(1))$; When in case 2 (prescalar has received 1 hit), then only if this spill had 0 hit, woudn't the prescalar fire. This probability is 1/2*(ppois(0)). So summing up these 2, we get tfreq(0). Similarly, we can find out all the tfreq(i). They are quite different from Poisson distribution.

Following is a table, for different orate vs. prate(the rate calculated from 0 trigger probability using naive Poisson distribution and assuming one q bin) at different prescalar values.

orate	prate	N	prate/orate
.2	.218	2	1.09
.4	.464	2	1.16
.5	.595	2	1.19
.7	.869	2	1.24
.9	1.158	2	1.29
.5	.656	4	1.31
.5	.685	8	1.37
.5	.692	16	1.38

Things are actually more complicated than the above, because:

A: If we put in dead time, the tfreq(0) might be affected. For example, say N=2, sometimes at one spill, we might actually get say 2 hits before the presvalar, but because of the dead time, only 1 is left, and it may not fire the prescalar. B: Not all the inputs into the mainors are prescaled by the same number.

Following is for run 2558, spec7. This is a positron run and Pion triggers are 98 percent of the mainors, prescalars are 2 2 2 1.

tfreq(0)**tfreq(1)**tfreq(2)**tfreq(3)**tfreq(4)**tfreq(5)**tfreq(6)

A 116378***81202****12713****1350*****92*******5********0

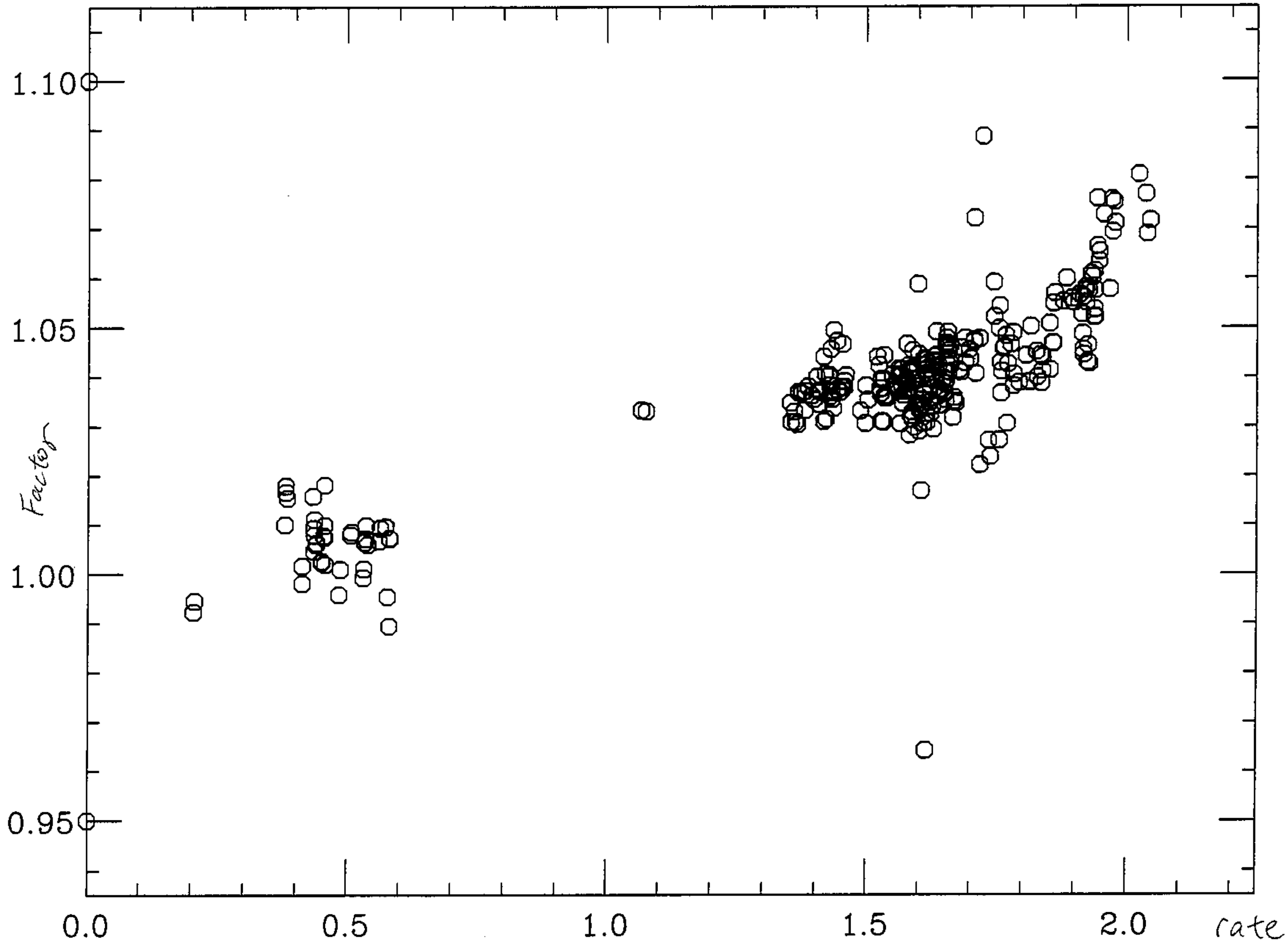
B 116378***69661****20852****4162*****623******75*******7

C 115673***70392****21009****4102*****513******42*******5

D 116378***84706****10210****452******10*******0********0

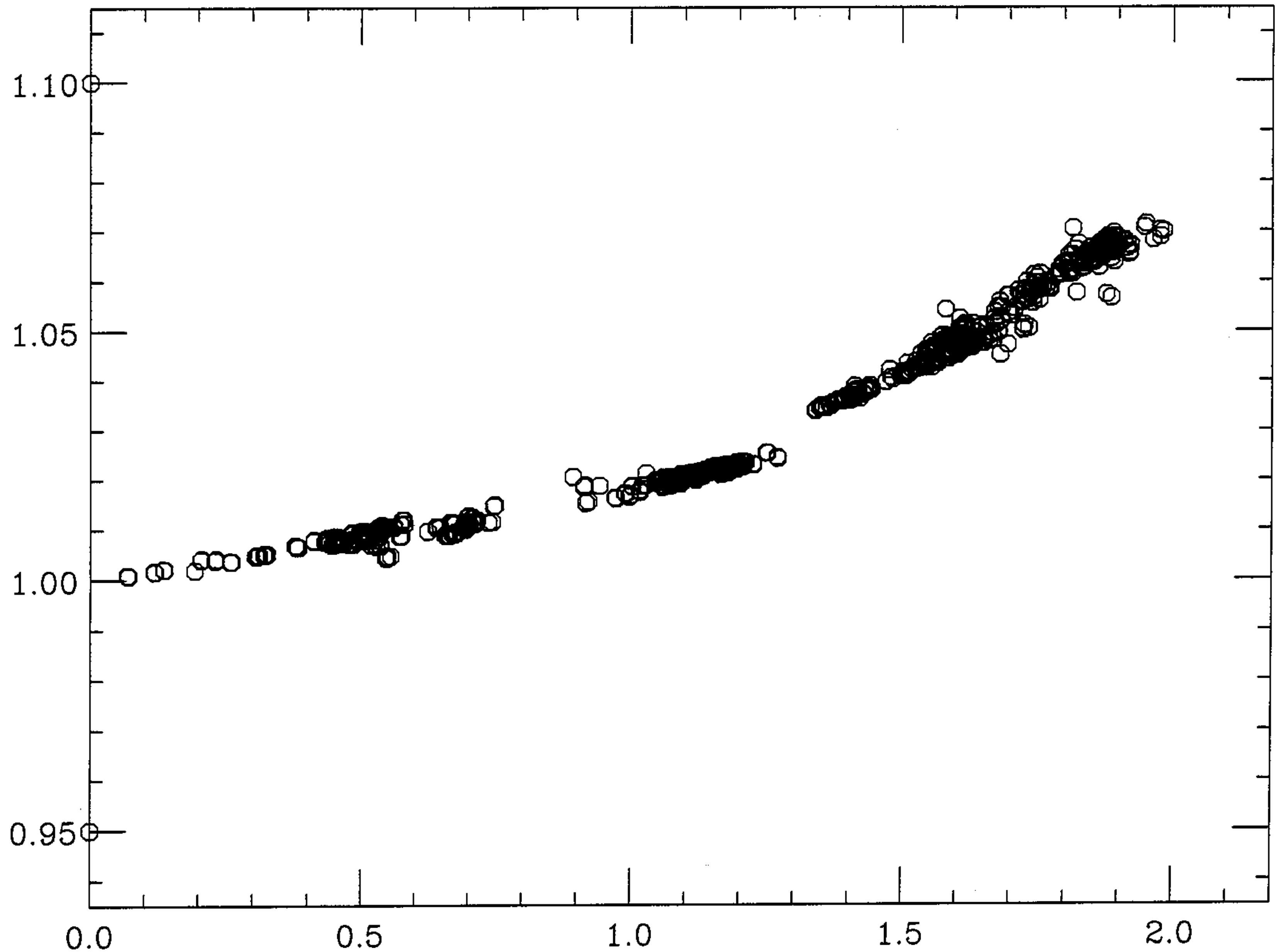
Note: A is observed distribution, B is naive poisson distribution using 0 trigger probability, assuming one q bin and no deadtime, C is Monte Carlo assuming a 32 ns deadtime, D is the "right" method, taking into account of a prescalar value of 2 (neglecting the 2 percent not prescaled main triggers in this run), assuming one q bin and no deadtime. We see that row D aggrees better than C or B.

"Good" runs for sp4 Method 1



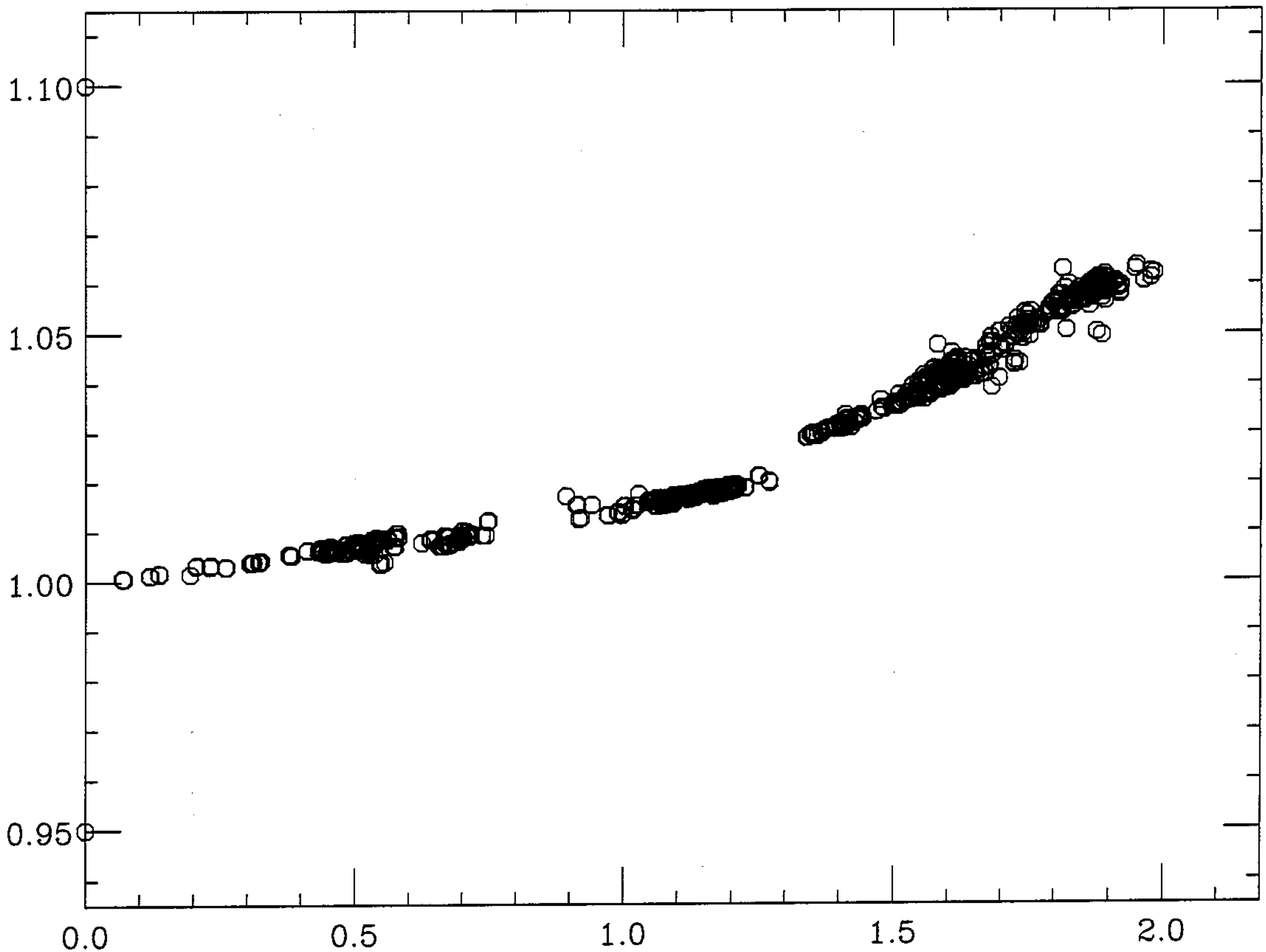
Method 2

dead time corrections vs. rate from deadjune,1800ns ,sp4



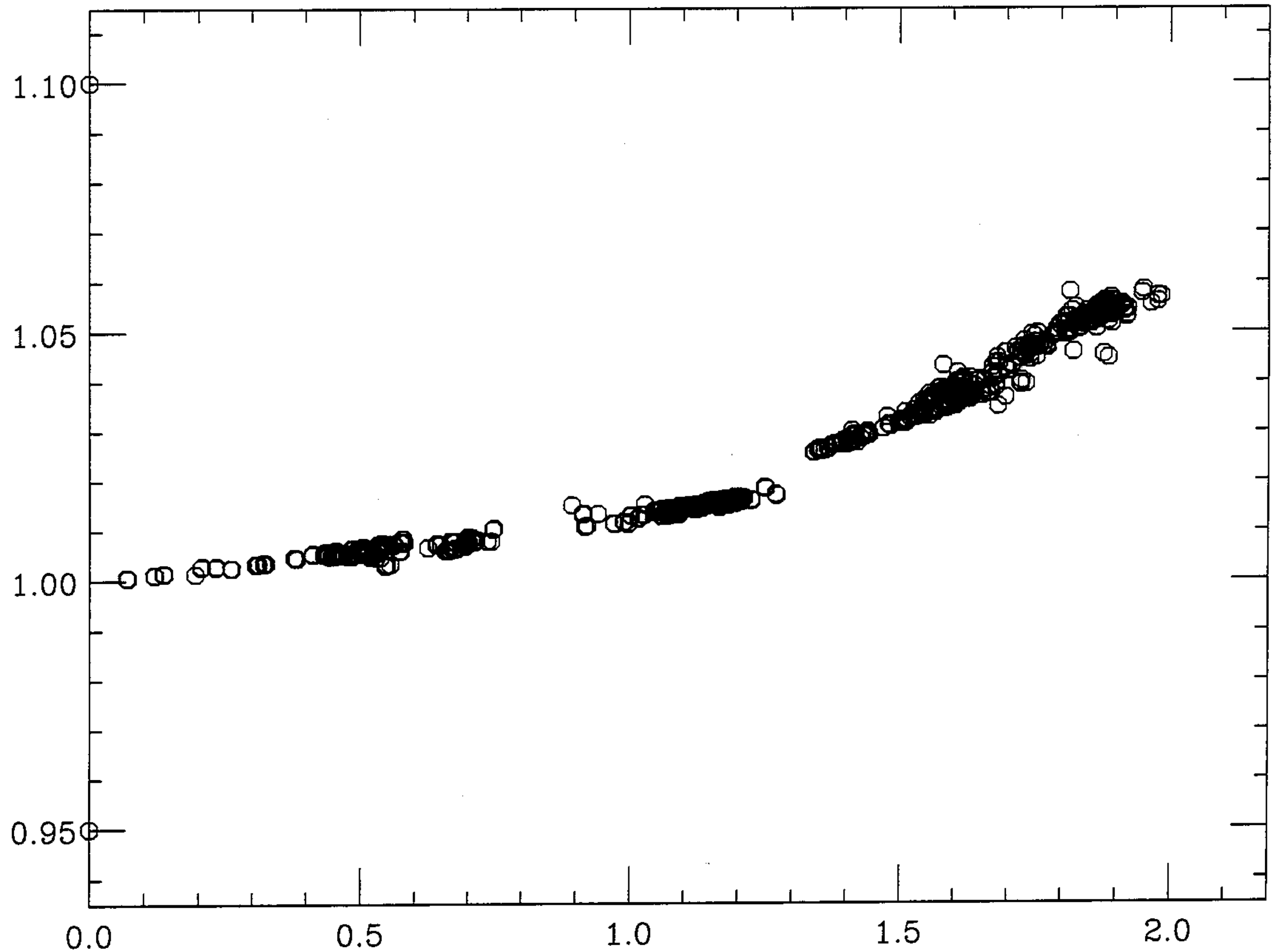
Method 2

dead time corrections vs. rate from deadjune,2200ns ,sp4



Method 2

dead time corrections vs. rate from deadjune,2600ns ,sp4



Method 2

dead time corrections vs. rate from deadjune,2200ns ,sp7