该题可以用贪心算法解决，即氮气加速器应该使用在公交车上面人数较多的时刻上面。设last[i]表示来到第i个景点的乘客最晚的时间，fast[i]表示车到达第i个景点的最早时间。

　　先考虑不用加速器的情况。可以直接递推求出答案：ans+=fast[man[i].end]-man[i].time，其中man[i].end为第i个人到达的站点，man[i].time是第i个人出发的时间。

　　接下来再来考虑使用加速器减少的时间。对于每个加速器，我们必须使这个加速器获得最大的效益，即使尽可能多的乘客旅行时间减一。如果我们在i到i + 1间使用加速器的话，那么到i + 1站的乘客的旅行时间都会减一，但是如果fast[i + 1]小于last[i + 1]的话(也就是车来太早了)，车就要在i + 1站等到t[i + 1]所有的乘客上车，在i + 1站以后下车的乘客的时间是一样的（车来早了没有用，要等），也就是说这个加速器对后面下车的乘客没有影响。设effect[i]为i站点最远能影响的车站，则有递推式：

　　effect[i] = effect[i + 1] //fast[i + 1] >last[i + 1] 不用等人 对以后的时间还会加速直到遇到要等人的车站为止

　　effect[i] = i + 1 //fast[i + 1] <= last[i + 1] 车来早了，要等人，使用加速器也没有效果，所以加速器的作用区间为从使用的那一站开始一直到某一站公交车需要等人为止，这个区间内的所有旅客都可以受益

那么，我们用一个加速器所能减少一个单位时间的乘客就是sum[effect[i]] - sum[i]，每次找出使这个最大的i减去节省的时间即可。

参考代码如下所示。