```
/* The original Traffic Control CPS Hybrid Program. */

\programVariables {
  R x1, v1, a1, t;  /* car 1 */
  R vsl, xsl;       /* traffic center */
  R B, A, ep;       /* system parameters */
}
/**
 * One lane, one car, one traffic center. Traffic center may issue speed limits at any time.
 Car needs up to ep time units to react (includes communication).
 * Car can brake and accelerate.
 * Checks if car complies with the speed limit after point xsl.
 */
\problem {
    (   v1 >= 0
        & vsl >= 0
        & x1 <= xsl
        & 2 * B * (xsl - x1) >= v1^2 - vsl^2
        & A >= 0
        & B > 0
        & ep > 0
      -> \[ (
                /* control car */
                (a1 :=
                -B)
                /* braking is always allowed */
                ++ (?xsl >= x1 + (v1^2 - vsl^2) / (2 * B) + (A / B + 1) * (A / 2 * ep^2 + ep *
                v1);     /* outside the speed limit do whatever you want, as long as you can
                still brake to meet the speed limit */
                    a1 := *;  ?-B <= a1 & a1 <=
                    A)
                ++ (?x1 >= xsl; a1 := *; ?-B <= a1 & a1 <= A & a1 <= (v1 - vsl) /
                ep);                    /* comply with the speed limit by not accelerating too
                much */

                /* traffic center, keep previous or set a new speed limit */
                (xsl := xsl; vsl := vsl)
                ++ (xsl := *; vsl := *; ?vsl >= 0 & xsl >= x1 + (v1^2 - vsl^2) / (2 * B) + (A /
                B + 1) * (A / 2 * ep^2 + ep * v1)); /* if we set a speed limit, the car must be
                able to comply with it, no matter how hard it currently accelerates */

                t := 0;
                /* dynamics */
                {x1' = v1, v1' = a1, t' = 1, v1 >= 0, t <= ep}
            )*
            @invariant(v1 >= 0 & vsl >= 0 & (v1 <= vsl | xsl >= x1 + (v1^2 - vsl^2) / (2 *
            B)))
        \] (x1 >= xsl -> v1 <= vsl)
    )
}
```