Forecasting Light-Duty Vehicle New Sales and Retention Rates

January 2023

Final Report

Prepared for the California Air Resources Board
and the California Environmental Protection Agency

CARB Agreement Number 21AQP018

Prepared by
Mark Jacobsen
University of California San Diego

Disclaimer

Acknowledgement

In addition to the very helpful support of CARB staff I wish to acknowledge research assistants Shunsuke Hori, Vafa Behnam, Erica Chuang, and staff in the UCSD economics department. This Report was submitted in fulfillment of CARB Agreement Number 21AQP018, "Forecasting Light-Duty Vehicle New sales and Retention Rates" by Mark Jacobsen, University of California San Diego, under the sponsorship of the California Air Resources Board. Work was completed as of January 31, 2023.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Number **Page**

ABSTRACT

We develop a new model of the light-duty vehicle population to improve the flexibility and realism of the Air Resources Board EMissions FACtor (EMFAC) model that estimates tailpipe emissions (i.e. pollution) from the vehicle sector. The vehicle population model here considers new-vehicle sales and retention rates of used vehicles, linking the two in a single economic system that can account for changes in travel demand and changes in regulation of the new-vehicle market. We also explore the potential role of inter-state vehicle trade in determining the response of sales and retentions to policy and demand conditions. Sample model runs demonstrate intuitive patterns in the way retention rates, and therefore the age profile of the vehicle stock, may be expected to change through time.

EXECUTIVE SUMMARY

Background

Modeling the pollution emissions of light-duty vehicles requires engineering details on emissions per mile, estimates of the accrual of miles per year, and estimates of the California vehicle population. This research focuses on vehicle population, improving the way that the sales of new vehicles and retention rates of used vehicles (which together determine vehicle population and age at any given time) are modeled.

The work here connects vehicle sales and retention to each other, and to overall travel demand, in an economically consistent way. It builds on work in the existing EMissions FACtor (EMFAC) model, allowing California Air Resources Board (CARB) to improve the flexibility and realism of its model of light-duty emissions. Better understanding the evolution of the population of new versus used vehicles is critical to estimating total pollution, especially as the typical new vehicle becomes even cleaner (perhaps even perfectly clean) relative to vehicles of various older vintages.

Methods

Households in the state meet their overall travel demand (a key input to the model here) through an evolving fleet of vehicles of different ages. Changes in travel demand, or in regulation affecting new vehicle sales (the second key input to the model here), alter both sales and retention rates through time. This study creates a computational model of these effects.

At its core, the model computes a dynamic equilibrium path where supply and demand of vehicles of each age are in balance. A force like reduced travel demand will lead to reduced sales of new vehicles and reduced retention of older vehicles in the fleet. Further, the reduced demand for used vehicles reduces the price people are willing to pay for them, which in turn reduces retention (i.e. supply) of vehicles of that vintage in the used car market. The equilibrium -- balancing supply and demand -- is necessarily also dynamic, meaning that the prices and choices in different time periods are all connected to one another. Choices in different time periods are connected because vehicles are assets (albeit relatively rapidly depreciating ones) that are bought and sold through time. The future value of a vehicle relative to its current value defines the depreciation portion of ownership cost, again calculated internally in the model. Retention rates, demand, and the asset model are all incorporated into a single economic framework grounded in standard theory.

The computable model takes inputs designed to relate directly to those in existing EMFAC tables and produces output (sales and retention paths) that may also be directly incorporated into EMFAC model runs.

Results

Sample runs of the model code demonstrate a range of intuitive patterns in sales and retention. For example, if travel demand slows over time, we see falling retention trimming vehicles out of the used fleet, first at the oldest ages and then at more intermediate ones. A model with static retention, as in current versions of EMFAC, will not project the removal of old vehicles since their stock is determined only by historical sales and the static survival curve. Another example is in the effects of regulation, where an increase in new vehicle costs is initially met with greater maintenance and retention of used vehicles (instead of sales), and then later with a gradual rebound in sales as repair costs rise in the used fleet. Throughout, overall travel demand in the state continues to be met and vehicle prices rise or fall to clear markets.

Conclusions

The model detailed here offers core improvements to the way EMFAC can model the vehicle population, and therefore to the final emissions estimates. In particular, the age profile of vehicles is made flexible, following the economics literature showing how vehicle retention and scrap rates connect closely to policy. Rather than maintaining a static retention rate through time, changes to new-vehicle policy can move retention up and down as individual households in California reoptimize their vehicle purchase decisions. Extending or shortening the life of used vehicles has important implications for emissions since used vehicles are generally more polluting, having been produced at a time when regulation on new vehicles was less stringent.

Two key expansions of the model here could allow important added insights. The first is to disaggregate vehicles at each vintage by fuel type, separately tracking zero emissions versus other (mainly gasoline) vehicles. While this requires new data and information on, for example, the ease with which households would be willing to substitute across vehicle types, it would allow examination of more detailed substitution patterns and subtleties of age profile changes across fuel types. A second expansion would be to model trade with the rest of the United States in a more detailed way. A simple linear model of trade is included here to allow analysis of potential trade patterns, but a fuller model could uncover more nuanced elements and allow prices in the rest of the country to react to changes in import or export demand from California.

SECTION 1.
INTRODUCTION

Modeling the pollution emissions of light-duty vehicles (LDVs) requires forecasting sales of new vehicles as well as forecasting the retention rates of used vehicles. Since used vehicles are particularly highly polluting (Jacobsen et al 2022), forecasts of retention rates are critical to estimating total pollution from the vehicle sector. This will become even more important as the share of electric vehicles grows among newer vintages: most of pollution emissions will start to depend on what the retention rates of older gasoline vehicles are forecast to be.

In recent versions of California Air Resource Board (CARB's) official emission inventory model for mobile sources EMission FACtor (EMFAC), the retention rates of used vehicles are held fixed at their historical average. New-vehicle sales and overall travel demand are forecast based on macroeconomic indicators and demographics.

Here we develop a computational equilibrium model that considers the interaction between vehicle retention rates and sales. It creates a dynamic forecast of the vehicle population, and we will abbreviate the model as "DVPop." The step taken here to model sales and retention rates together creates a more consistent picture of the vehicle population and an improved forecast of the likely age profile of vehicles going forward. The DVPop model considers two key forces that influence sales and retention rates:

i) Overall demand for light duty vehicles. When overall demand for vehicles goes up this will tend to increase new-vehicle sales and, depending on the speed of change and other factors, also increase typical retention rates to "produce" more used vehicles. If overall demand goes down, sales and retentions also fall.

ii) New-vehicle policy. When policy makes new vehicles more attractive this will increase new vehicle sales and decrease retention rates. If new vehicles become less attractive this will decrease sales and increase retention rates. Unlike effect (i) where sales and retentions move together, changes in new-vehicle policy tend to move sales and retentions in opposite directions (Gruenspecht 1982, Jacobsen and van Benthem 2015).

The model here, as with existing versions of EMFAC, abstracts from changes in the intensity of use of vehicles, i.e. mileage accruals per vehicle. DVPop also uses the fixed profile of accruals from EMFAC, which captures the feature that newer vehicles tend to be driven more intensively, but does not attempt to forecast changes in the age profile of accrual rates through time. Notes on the potential for doing this with future versions of the model appear in the *Recommendations* section below.

The value of an equilibrium model in this setting comes from the fact that consumer demand links all new and used vehicle markets together into a single system. Consider the following example: Household A, that might otherwise have held on to its 5-year-old vehicle a little longer, decides to trade it in for a new one in order to take advantage of a new subsidy program offered by the state. There is now a surplus of age 5 vehicles in the California market, reducing their price a little. Household B, which would like to buy a used vehicle and would normally have chosen a 7-year-old one, now chooses the 5-year-old one instead. This creates a small surplus of 7-year-old vehicles and lowers their price. This chain continues. At some point there is a vehicle dealer with a car in need of repair, for example suppose it has an 18-year-old car with a broken transmission, that it no longer makes financial sense to repair because of the lower values throughout the used market. The dealer decides to scrap the vehicle. This lowers the supply of 18-year-old vehicles, preventing their price from falling as far. In versions of the model that allow trade to and from California, dealers can also decide to move used vehicles in or out of the state in response to changes in their value, further adjusting supply. An equilibrium occurs where prices at all ages are such that the market clears and there is no excess supply or demand.

The equilibrium over time is too complex to simulate at the level of individual consumers and vehicle models and so we look at each age of vehicle in aggregate. We also aggregate over regions of California, needed because most of the used vehicle market operates at the scale of states or countries rather than that of cities or counties. EMFAC contains new vehicles and 44 ages of used vehicles, so in the demand system here we include 45 corresponding choices that a representative consumer may make. Each of 44 age categories of used vehicles may be retained or scrapped as the model moves forward in time, so there are 44 aggregate supply decisions. In a dynamic sense, there are also 44 state variables describing the system, making it a computationally challenging problem. The supply of new vehicles is assumed to be exogenous. Prices of new vehicles are determined by larger forces in the global market, combined with local policy constraints, and enter the model as a set of input parameters discussed in detail in Section 2.

Section 2 of this report also lays out the model structure, including all of the key equations used in solving the equilibrium system and the numerical inputs needed for calibration and scenario analysis. Section 3 describes results from a series of demonstration equilibria where scenarios involving changing travel demand forecasts and new-vehicle policy implementation are developed. Section 4 discusses caveats and interpretation of inputs. Section 5 offers a summary and general conclusions. Finally, Section 6 details recommendations for further study and extension of the model.

SECTION 2.
METHODS

The model is a numerical analysis consisting of supply and demand equations and a time component. Together, the equations in the structure below define an equilibrium that can be solved for using a nested set of search algorithms. The resulting equilibrium describes the evolution over time of vehicle sales and retention rates, key inputs to the EMFAC model. Using a structure like this one to jointly predict sales and retention leads to improvements in the theoretical consistency of the EMFAC model and can reduce reliance on re-scaling to rationalize outputs.

2.1 Model Structure

The structure here is based on work in Jacobsen et al (2021), which models elasticities of demand and scrappage between new and older vehicles due to changes in regulation, with two added flexibilities. The first of these is that the present model can account for changes in overall vehicle demand, for example simulating the pattern of increase in sales and retentions if overall vehicle demand in California rises. Second, it can account for trade in used vehicles, meaning that retentions can change both through changes in vehicle scrap rates and through changes in imports or exports.

2.1.1 Notation

a indexes vehicle age. Varies between 0 (new) and an upper bound A set to 44.

t indexes time. Varies between 0 (base year) and an upper bound T (set to 60 in the examples shown here).

$q_{a,t}$ number of vehicles of age *a* in equilibrium (which is defined such that supply and demand are equal in all time periods simultaneously) at time *t*, so $\sum_a q_{a,t}$ is the total vehicle stock at time *t*

$q^D_{a,t}$ aggregate number of vehicles of age *a* demanded by the representative consumer at time *t*

$q^S_{a,t}$ aggregate number of vehicles of age *a* supplied at time *t* (depends on the equilibrium vehicle stock at time *t*-1 and on expectations about demand in future time periods)

$s_{a,t}$ probability that a vehicle is scrapped between ages *a-1* and *a* in time *t*

$h_{a,t}$ average repair cost paid to keep a vehicle of age *a* from being scrapped in time *t*

$i_{a,t}$ net imports to the state of vehicles of age *a* at time *t*

$n_{a,t}$ net retention rate of vehicles of age *a* at time *t*. Combines the retention of in-state vehicles, $1 - s_{a,t}$, with imports, $i_{a,t}$. Net retention is defined by $n_{a,t} \equiv \frac{q_{a,t}}{q_{a-1,t-1}}$

$p_{a,t}$ asset value (price if being sold in the market) of a vehicle of age *a* at time *t*

$r_{a,t}$ ownership cost of a vehicle of age *a* at time *t*. Includes depreciation and repairs, $h_{a,t}$

2.1.2 Demand

The demand system in the model represents the quantity of vehicles of each age that would be demanded under any given set of vehicle prices. The aggregation up to vehicle age is due primarily to lack of information about substitution elasticities across fuel types and vehicle classes, and also due to computational limitations that would likely become important if the choice set is expanded too far. Probably the most logical next extension, discussed in the *Recommendations* section below, is to expand the model to two explicit fuel types. Demand elasticities between fuel types and ages could be selected (perhaps as scenarios) and then changes in sales and retentions could be tracked separately for gasoline and electric vehicles.

The demand system over vehicles of different ages is written in terms of the ownership costs associated with choosing each vehicle age, $\boldsymbol{r} \equiv [r_0, r_1, \dots, r_A]$. Modeling ownership cost as the variable of interest to the consumer follows the approach in Bento et al. (2009) and is important when allowing choices to be made between new and used vehicles. For the literature that examines choice among different new vehicle models (excluding used vehicles from the choice set), purchase price is the more typical measure and produces equivalent elasticities.[1]

To represent this large system of demand elasticities flexibly we follow Deaton and Muellbauer (1980). Their demand system approximates any utility-consistent setting using a set of $n(n-1)/2$ free elasticity parameters (where n in this setting is 46: new vehicles, used vehicles of ages 1 through 44, and an outside option). Demand for vehicles is given by:

[1] The equivalence in the case of a new-vehicle-only analysis holds when residual value after a typical holding period is a constant fraction of the purchase price. Then, a 1% increase in the purchase price equates to a 1% increase in the ownership cost (i.e., purchase price less residual value), so the relevant demand elasticities are identical.

$$q^D_{a,t}(\boldsymbol{r_t}) = \frac{M_t}{r_{a,t}}\left(\beta_a + \sum_{\hat{a}=0}^{A} \theta_{a\hat{a}} \ln\left(r_{\hat{a},t}\right)\right) \tag{2-1}$$

where $\boldsymbol{r_t}$ is the vector of prices for all vehicle ages. The set of β parameters determine initial demand shares of vehicles of different ages and are calibrated to baseline data. The $\theta_{a\hat{a}}$ parameters control derivatives with respect to prices and can be mapped directly into any matrix of own- and cross-price elasticities. These may be taken from the economics literature which includes a range of studies estimating demand elasticities for vehicles. See Jacobsen et al (2021) for a survey. Finally, aggregate spending in the demand system, M_t, is an input parameter to the present model: it can be chosen to reflect any given path of predicted travel demand in the state. For example, choosing a constant growth rate for M_t suggests constant overall demand growth and produces a simple steady state in the equilibrium system here. Setting the growth rate of M_t to change through time, for example reflecting slowing or accelerating population growth in the state, leads to a transition path in the vehicle market as quantities at different ages adjust in order to meet changes in travel demand. See Results section for an example.

Deaton and Muellbauer (1980) show how basic restrictions on $\theta_{a\hat{a}}$ from utility theory (symmetry, invariance to units, and adding up) translate easily into their setting. The choice of this demand system has two main advantages. First, it allows for an intuitive set of inputs (elasticities directly, as opposed to structural parameters) allowing easy exploration of alternative scenarios. And second, all general theoretical restrictions on utility are readily satisfied.

2.1.3 Supply

The price of new vehicles is taken as an input and new-vehicle supply is assumed to be perfectly elastic at that price. That is, California new-vehicle buyers purchase new vehicles on a global market and changes in state-level demand for new vehicles (at least changes of the magnitudes we model) are not large enough to substantially move new-vehicle prices. The input price path of new vehicles can vary over time to reflect regulation or other expected changes in the vehicle market: for example prices of the typical new vehicle purchased may be expected to rise because of electric vehicle mandates, or fall because of subsidies. New vehicle price inputs are discussed in the calibration and inputs subsections below.

The supply of used vehicles is given by the number of vehicles of that vintage present in the previous time period times a retention rate, $n_{a,t}$. If there were 200 5-year old vehicles last time period and retention is 0.9, then there will be 180 6-year old vehicles in the current time period. The retention rate depends on rates of scrap versus repair when vehicles are in accidents or have mechanical failures, and on rates of export or import of used vehicles. These quantities both depend on vehicle values.

We discuss the scrap versus repair decision first: Jacobsen and van Benthem (2015) show that a straightforward scrappage function can be built from an underlying distribution of repair cost shocks (e.g., costs from mechanical failures and accidents) and the assumption that scrap occurs when the realized repair shock exceeds the value of the vehicle. An underlying distribution of cost shocks with positive support corresponds to a scrap function $s_{a,t}(p_{a,t})$ that slopes downward as vehicle values increase. To keep our equilibrium calculation as straightforward as possible we considered a constant elasticity scrappage function taking two parameters: one parameter that scales scrappage levels in the baseline, and another that sets the scrappage elasticity. Following Jacobsen and van Benthem (2015):

$$s_{a,t} = b_a \cdot (p_{a,t})^{\Upsilon} \tag{2-2}$$

where b_a is a baseline scale term and $\gamma < 0$ is the scrappage elasticity (defined as the percentage change in scrappage for each 1% increase in vehicle value).

The second component of net retention is imports. We normalize the import term to zero in the baseline such that values of $i_{a,t}$ in the model represent changes: positive values of $i_{a,t}$ indicate either more imports (if the baseline world has net imports at age *a*) or fewer exports (if the baseline world has net exports at age *a*). With data on trade patterns in and out of California in the baseline, the relative changes reported in $i_{a,t}$ could be converted into absolute levels of import or export.

Imports are modeled with a simple linear function, though this could be easily modified if estimates of a non-linear shape become available. Imports are given by

$$i_{a,t} = \mu_a q_{a,0}\left(p_{a,t} - p_{a,0}\right) \tag{2-3}$$

where the parameter μ_a controls the rate of change in import or export of a given vehicle age for each dollar increase or decrease in price relative to the baseline. Without a formal analysis of vehicle trade in and out of California setting this parameter is challenging. However, exploring scenarios with zero (no trade) and large (nearly free trade with the rest of the country) values may provide useful bounds. Intermediate positive values can also be chosen to explore cases where some fraction of an increase in demand gets met with imports and some fraction with reduction in local scrap rates.

Overall used vehicle supply, as a function of used vehicle values, can be written as:

$$q^S_{a,t} = \left(1 - b_a\left(p_{a,t}\right)^{\gamma}\right)q_{a-1,t-1} + i_{a,t} \tag{2-4}$$

2.1.4 Vehicle asset values and ownership costs

The portion of annual ownership cost relevant to the scrap decisions and equilibrium effects studied here primarily reflect repair costs (to avoid scrap) and losses in asset value through time. Fuel and other accrual-related costs don't vary with the choices in the present model (since accruals are fixed conditional on age) though could in principle be brought in should forecasts of changes in accruals be integrated.

If the world is in a steady state, then losses in asset value (the depreciation portion of ownership cost) can be determined simply by looking at history: if new vehicles have tended to lose 40% of their value in the first 3 years, then a new vehicle purchased this year should behave the same way. When changes are occurring because of a policy phase-in, or anticipated aggregate demand shifts, this is no longer the case. If a policy change is reducing new-vehicle sales, for example, then dealers and buyers can reasonably expect that used vehicles will become a scarcer, and therefore more valuable, a few years from now. Expected depreciation and ownership costs are therefore smaller. The equilibrium model here is designed specifically to capture these effects, creating a time path consistent with utility maximization over time.

The depreciation cost of a vehicle at any given age and point in time can be written as:

$$r_{0,t} = p_{0,t} - \frac{1}{1+\delta}E_t\left[(1 - s_{1,t+1})(p_{1,t+1} - h_{1,t+1})\right]$$
$$r_{1,t} = p_{1,t} - \frac{1}{1+\delta}E_t\left[(1 - s_{2,t+1})(p_{2,t+1} - h_{2,t+1})\right]$$

...

$$r_{A,t} = p_{A,t} \tag{2-5}$$

where $E_t[\cdot]$ refers to the expectation held at time t about future vehicle values. In equation (2-1), the 1-year ownership cost of a new vehicle, for example, is its new vehicle price less its residual value (the expected value of a 1-year-old vehicle next year, net of scrap and repair). When agents are forward-looking and maximizing utility we will have that $E_t\left[p_{a,t+1}\right] = p_{a,t+1}\ \forall\ a, t$.

2.1.5 Dynamic Market Equilibrium

For ease of notation, we first observe that demand at time t, expressed in equation (2-1) as a function of $\boldsymbol{r}_t$, can be rewritten as a function of $\boldsymbol{p_t}$ and $E_t[\boldsymbol{p_{t+1}}]$ by using equation (2-5). Bold type indicates that prices are a vector across ages.

An equilibrium over a finite horizon T is defined as a set of vectors $\boldsymbol{p_t}$, for t between 1 and T, such that the following conditions hold:

i) New-vehicle quantity in each time period is given by demand (given elastic supply of new vehicles):

$$q_{0,t} = q^D_{0,t}(\boldsymbol{p_t}, E_t[\boldsymbol{p_{t+1}}]) \tag{2-6}$$

ii) Used-vehicle quantities for all $a \geq 1$ are such that demand and supply are equated at each time t:

$$q^D_{a,t}(\boldsymbol{p_t}, E_t[\boldsymbol{p_{t+1}}]) = q^S_{a,t}\left(q_{a-1,t-1}, \boldsymbol{p_t}\right) \tag{2-7}$$

where $q_{a,0}$ (the incoming vehicle stock from before the model starts running) is given by the long-run age profile of vehicles implicit in historical retention rates

iii) Expectations are correct such that:

$$E_t[\boldsymbol{p_{t+1}}] = \boldsymbol{p_{t+1}} \; \forall \; t \tag{2-8}$$

2.1.6 Computation

We use the following algorithm to find a series of price vectors that satisfy all of equations (2-6) through (2-8) above:

1) Set the starting value of expectations equal to the baseline price vector.

2) Solve for the individual time period equilibria from $t = 1$ to T sequentially (because supply at time t depends on quantities from $t - 1$).

3) Use the results from Step (2) to assign a new set of price expectations.

4) Iterate Steps (2) and (3) to a fixed point.

Step (2) ensures that equations (2-6) and (2-7) hold in every time period given expectations. Steps (3) and (4) ensure that equation (2-8) holds.

2.2 Inputs and Base Calibration

Table 2-1 lists the inputs needed to calibrate the elasticities in the model and establish the steady-state equilibrium in the base year.

Table 2-1. Model Inputs

Parameter	
δ	Discount rate
κ	Baseline growth rate of the vehicle stock
γ	Scrappage elasticity
n_a	Baseline retention rates
p_a	Baseline used vehicle asset values
$\theta_{a\hat{a}}$	Substitution parameters in demand system
μ_a	Rate of change in net vehicle imports or exports as California vehicle prices rise or fall

2.2.1 Input Sources

δ: We use a real discount rate of 3%, effects of changes in the rate are small since this only influences the internal deprecation calculation. The discount rate does not need to be adjusted for inflation; all values are in constant dollars throughout.

κ: We use a baseline growth rate of 0.12% per year, following long run trends in EMFAC2021.

γ: The scrappage elasticity is set to -0.7, following Jacobsen and van Benthem (2015).

n_a: Baseline retention rates are taken from EMFAC2021. They are a long run average of retention rates in the fleet. These values determine the baseline age profile of the fleet and also the level of new sales needed to replace scrapped used vehicles. They form the core of the base year calibration for the dynamic model. The scale parameters in the demand function, β_a, are then calibrated such that the demand system reproduces exactly the long run age distribution implied by these retention rates.

p_a: Baseline used vehicle asset values are taken from Jacobsen and van Benthem (2015). These are national data; comparable values for California are not regularly published. The effects in the dynamic model depend mostly on price changes, as opposed to levels, so these values play a relatively small role in the results.

$\theta_{a\hat{a}}$: Demand elasticities between different aged vehicles and between vehicles and the outside good (travel via other modes and other spending) can be calibrated to a range of studies in the economics literature. See Jacobsen *et al* (2021) for a survey. The elasticities chosen for the examples in this report are -0.8 for the aggregate own-price elasticity of new vehicle demand[2] and -0.05 for the aggregate demand for private vehicles relative to other transportation options. The choice of -0.05 is at the lower end of the plausible range from the literature and reflects a world where very little substitution away from private cars is feasible. Or equivalently, where any substitution away from private vehicles is driven by exogenous factors rather than the types of policy and price changes modeled in EMFAC.

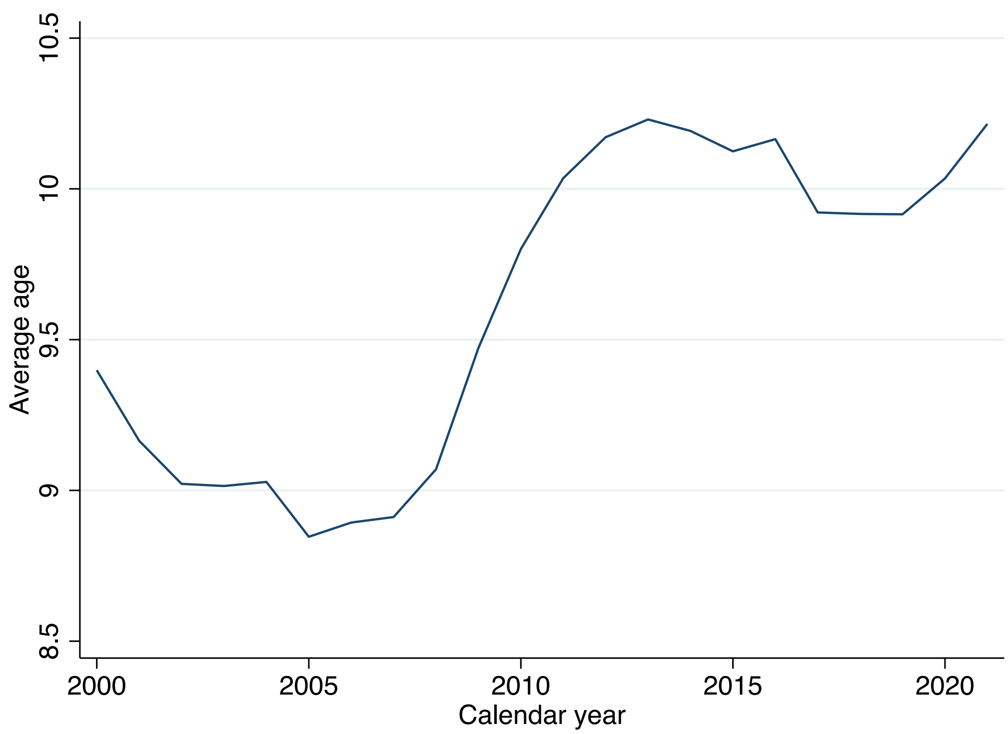
μ: Little analysis exists on trade patterns in used vehicles across U.S. states, and we do not know of any studies that estimate the sensitivity of trade to vehicle price or tax differences across states (the effect captured in μ). Work in Davis and Kahn (2010) examines international trade. Setting $\mu = 0$ may best reflect prior versions of EMFAC, where the California vehicle market is assumed to be self-contained. Choosing higher values for μ could provide interesting scenario analyses in the presence of more open trade, for example cases where relatively highly polluting used vehicles can flow into California, or where EVs first sold in California can flow out over time. In all cases imports and exports in the base case are normalized to zero, so results are expressed relative to historically typical trade flows. This means that trade in the base year of the mode (t=0) will always be reported as zero; trade values in later years are changes resulting from the input changes in travel demand and policy.

[2] The value -0.8 lies in the middle of the range from a long literature on automobile demand, including work in Berry, Levinson, and Pakes (1995, 2004), Goldberg (1998), Bento et al (2009), Knittel and Metaxoglou (2014), Dou and Linn (2020), and Leard (2021).

2.2.2 Calibration of retention rates and baseline vehicle age profiles

The dynamic model takes as input historical average retention rates at each age, a total vehicle stock at time 0, and a historical average growth rate in the stock. This is enough to create a baseline pattern of ages in the fleet and to compute the sales needed to maintain stable growth in the stock. While in any given year retention rates, sales, and the stock all move around due to shocks to the system, patterns in retention and stock size over longer horizons appear relatively stable. Patterns of growth over 40 years in the stock are shown in Figure 4-2. Patterns in average age over 20 years are shown here in Figure 2-1. To the extent a changing trend in retentions is anticipated going forward (maybe newer vehicles, for example those made with electric drive trains, will last longer, or less long, than vehicles have in the past) this could potentially be added as an input. However, given the historical data, assuming stability in lifetimes and growth in the stock seems to be a neutral starting assumption.

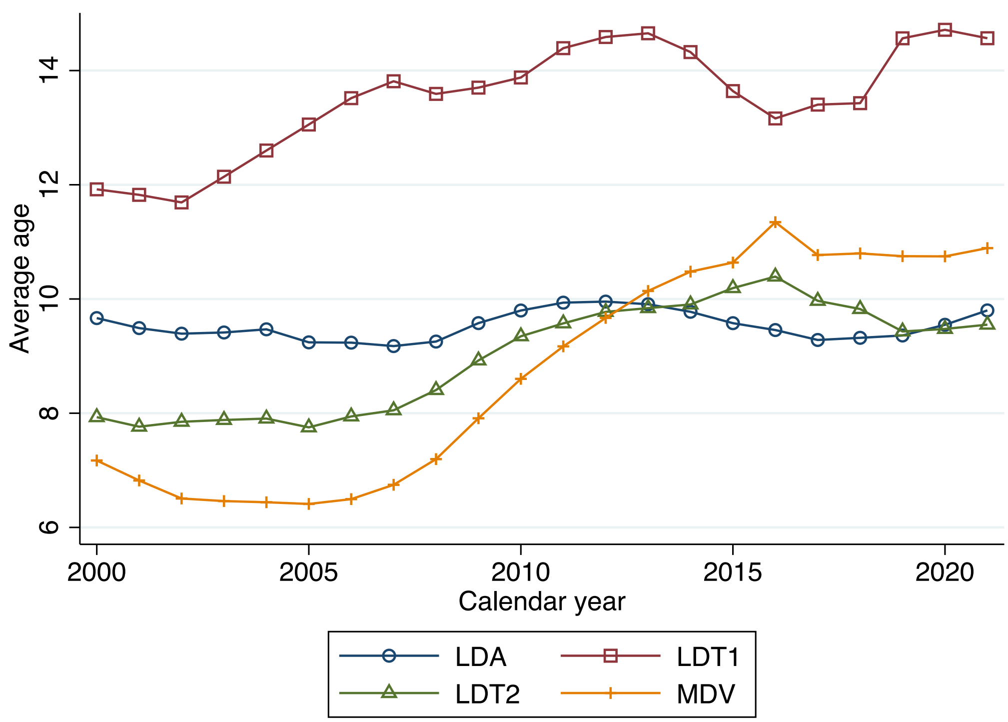
Figure 2-1. Average Vehicle Age Among Those Less Than 45



Notes: Data source is *dmv_population.sql* as provided by CARB. Calculation here defines age to be 0.5 when model year and calendar year match. Maximum age is 44.5. Includes all vehicles with the *veh_class* data field equal to LDA, LDT1, LDT2, and MDV.

Figure 2-1 shows that average vehicle age surged in the years following the great recession but has been relatively stable since 2012. The future could potentially hold a return to the younger vehicles of the early 2000s, or maybe a continuing trend toward even older vehicles. Some evidence of recovery toward younger ages was visible in 2017 and 2018, though this was undone by another upward movement in age visible in 2020 and 2021, likely the result of new-vehicle supply chain disruptions due to the SARS-CoV-2 pandemic. The simplest starting assumption seems to be that average age will stay approximately the same in the long run (absent policy changes).

Figure 2-2. Average Vehicle Age By Class Among Those Less Than 45



Notes: Data source is *dmv_population.sql* as provided by CARB. Calculation defines age to be 0.5 when model year and calendar year match. Maximum age is 44.5.

Decomposing by vehicle class, we see that most of the increase in average age between 2008 and 2012 came from the MDV and LDT classes. Importantly, the relative stability in average age since 2012 still continues to hold when disaggregating by class; it is not that case that some classes are going up in age and others down in recent years. This is reassuring in the sense that an aggregate model, merging the classes together, will not be masking class-level trends. The absolute differences between classes (LDT1s tend to have much higher ages than other classes) will be preserved when applying the projected changes in aggregate retention here proportionally to the class-specific retention factors used in EMFAC.

2.2.3 Calculation of Retention Rate Inputs

Retention rates for the dynamic model are calculated as in EMFAC2021, but using data aggregated to the state level (since the used vehicle market operates at the state level, or an even larger scale, the relevant equilibrium is also at a large scale).[3] As an example, retention from the DMV vehicle population data is calculated as:

$$n_{\tilde{a}=10,t=2010} = \frac{pop_{my=2000,t=2010}}{pop_{my=2000,t=2009}}$$

where t is calendar year, my is vehicle model year, and $\tilde{a}$ is vehicle age defined as $\tilde{a} = t - my$. *pop* is the DMV registered vehicle population in *ldv_activity_total_wrule.sql* as provided by CARB. Note that the retention rate at age 45 is 0 by construction; we do not include any model year 1965 vehicles in calendar year 2010 such that, for example, $n_{\tilde{a}=45,t=2010} = pop_{my=1965,t=2010} = 0$. Retention rates at age 0 are undefined and not needed for the model.

An important detail has to do with the definition of age in the EMFAC retention rate analysis (the difference between model year and calendar year) versus the definition of age in DVPop (the amount of time that has passed since vehicle purchase). These are similar, but not identical, quantities since the October DMV snapshot means that some vehicles of the previous model year (11% on average in the data) were sold less than 12 months ago. These vehicles will therefore be assigned $\tilde{a} = 1$ when in fact they have been on the road for less than a year.

The calculation performed for EMFAC retention analysis for age 1 is:

$$rr_{a=1,t=2010} \frac{pop_{my=2009,t=2010}}{pop_{my=2009,t=2009}}$$

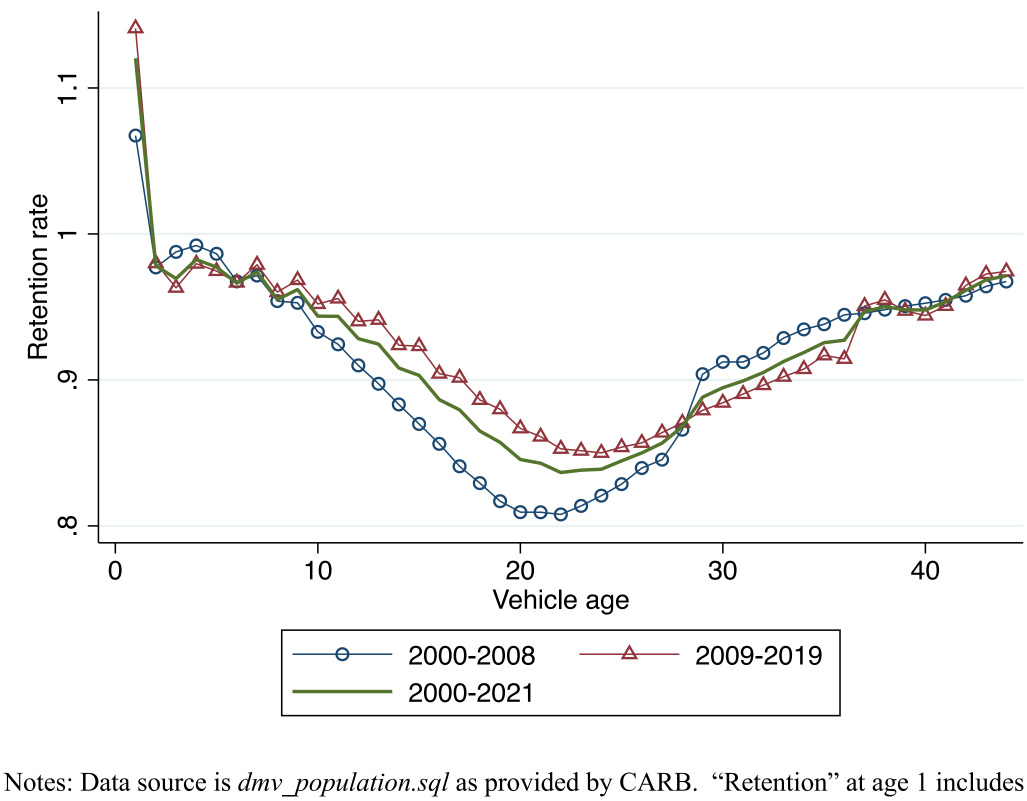
This typically results in retention appearing to be larger than 1 for 1 year old vehicles, when in fact what is happening is that some model year 2009 vehicles were sold at the end of 2009 and therefore appear in the numerator but not the denominator of the above fraction. Translating back and forth between $\tilde{a}$ and a is fairly straightforward (it amounts to shifting 11% of the

[3] In the calculations below we use the whole vehicle population, across all regions, to calculate aggregate retention. This is similar to taking a weighted average of retention across regions. An unweighted average of retention across regions produces larger values for average retention because small regions in California tend to have older vehicles and therefore higher retention rates.

vehicles in each model year bucket to a younger calendar age) and helper routines *q_to_myq()* and *myq_to_q()* are included in the DVPop module to make this easier. The model can output quantities either by calendar age (in variable *q*) or by model year age (in variable *myq*). The corresponding retention rates by calendar age are in variable *nr* and by model year age in variable *nrmy*. Similarly, the model can read in retention rates using either the calendar time or model year definitions of age.

Figure 2-3 displays aggregate retention rates using the model year definition (so they are larger than 1 at age 1). Note the sharply higher retention rates in the later part of the sample (the great recession and its aftermath), corresponding to the surge in average age shown above. The "zig-zag" pattern in retention rates is most likely the result of biennial inspection programs.

Figure 2-3. Aggregate Light-duty Vehicle Retention Rates 2000-2021

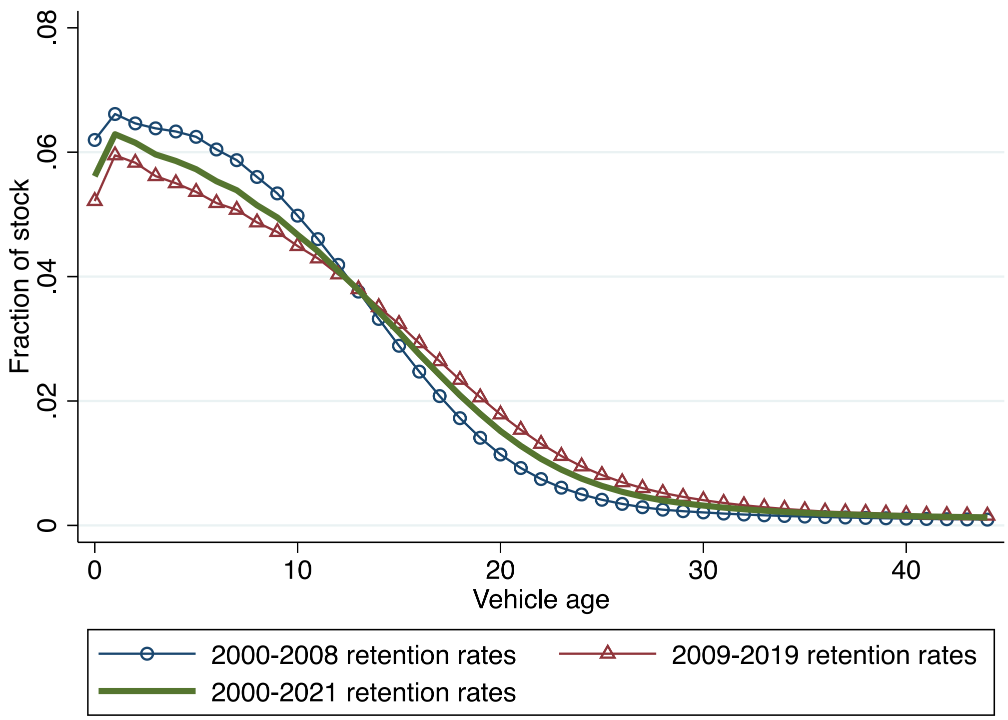


Notes: Data source is *dmv_population.sql* as provided by CARB. "Retention" at age 1 includes additional sales of model year *x* vehicles in calendar year *x*+1 and so typically exceeds 1.

Any given set of retention rates will imply, over time, a long run average profile of ages in the vehicle stock. Below we examine the age profiles (effectively, cumulative survival patterns) that correspond to different calibrations of the baseline retention rate.

Figure 2-4 maps the three sets of retention rates from Figure 2-3 into their long run age profiles. The average ages of vehicles are 8.86 (using 2000-2008 retention rates), 10.47 (using 2009-2019 retention rates) and 9.78 (using 2000-2021 retention rates).

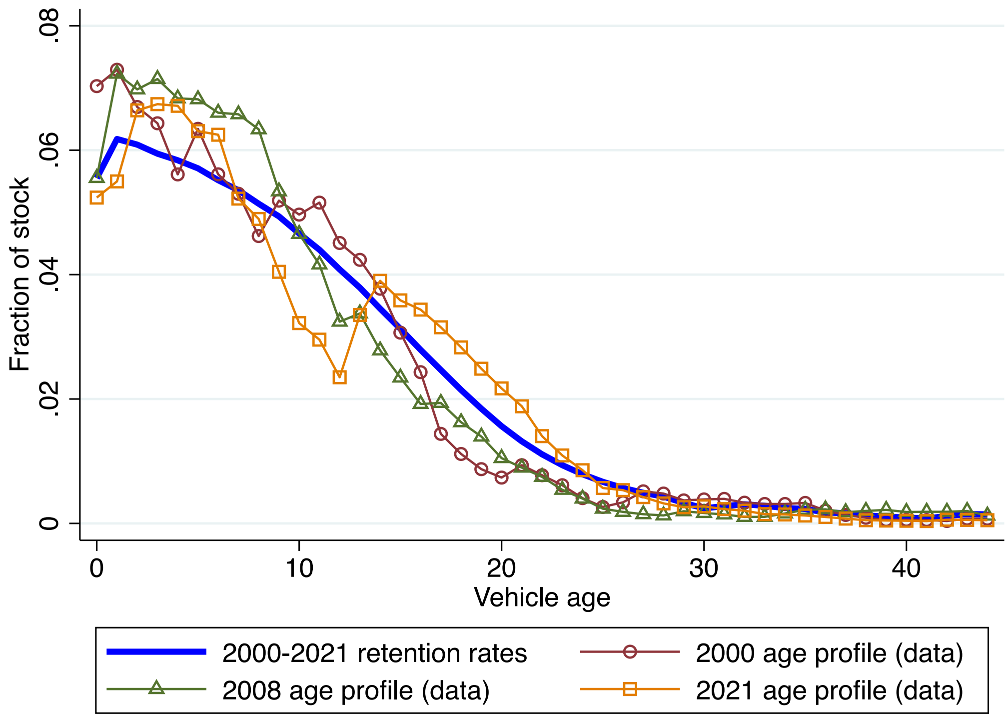
Figure 2-4. Age Profiles Implied by Different Samples of Retention Rate Data



Notes: Computed using data in Figure 2-3.

Figure 2-5 compares the long run age profile implied by 2000-2021 retention rates (which are equivalent to the EMFAC2021 projection for 2050) with data for years 2000, 2008, and 2021 as examples. The average ages in the data are 8.90 (2000), 8.57 (2008), and 9.72 (2021). The very close match between the EMFAC2021 long run projection and the calculations done in the DVPop model using retention rates provides reassurance that the assessment of retention rates done here matches that in prior work for EMFAC.

Figure 2-5. Aggregate Light-duty Vehicle Retention Rates 2000-2021



Notes: The line "2000-2021 retention rates" computes the stable vehicle stock using the retention rates shown in Figure 2-3 and produces a long run age distribution equivalent to the 2050 values for EMFAC2021 in *ldv_activity_total_wrule.sql*. Data for the 2000, 2008, and 2021 age profiles is from *dmv_population.sql*.

2.3 Inputs for Scenario Analysis

Once the model is calibrated with base data and elasticities, various scenarios may be evaluated. The default scenario is simply constant growth in the size of the California vehicle market and no change in the cost of new vehicles. This results in a steady state: the model will forecast no change in retention rates and steady growth in vehicle sales at the same rate as the growth in overall vehicle demand.

Two sets of inputs to change the forecast scenario are available. The first input allows the growth rate in overall demand for travel and vehicles to move away from the historical pattern. The second alters the average cost of new vehicles over time. The resulting outputs will show the connected evolution of vehicle sales and retention rates under the assumptions of the scenario. These two input vectors are discussed in detail below:

2.3.1 Changes in Travel Demand

This input provides a way to enter predictable shifts in the underlying patterns determining travel demand. The input is labeled *vmtgrowthvec* in the dynamic model. For example, one key determinant of travel demand is human population. If demographic forecasts suggest California's population growth will accelerate or decelerate, then travel demand can be forecast to do the same. Another example would be changes in income growth rates over time. If California's period of rapid economic growth is expected to fade, for example, to the more modest growth typical of the rest of the U.S., this would slow travel demand over time (Sivak, 2013). Other factors that this input could be used for include things like forecasts of large-scale public transport investment in the state, high-density housing regulations, or telecommuting technology growth. These things can be expected to replace commutes by private car with digital or transit commutes, again reducing California private vehicle travel demand over time. Other factors like improved self-driving technology could have the opposite impact, increasing the demand for travel in light-duty vehicles.

Short-term shocks, like the timing of recessions, gasoline price movements, pandemic conditions, and so on are by their nature extremely difficult to predict. Should predictions of these things be available they could in principle also be included in the input variable controlling overall demand. An argument against including these sorts of shocks is that they are much more likely to be absorbed into accrual changes as opposed to vehicle demand changes: vehicle owners know that these shocks are transitory and so will try to wait them out. A change in human population, or a new subway station, on the other hand, are quite likely to permanently add or remove commutes, and so are much more likely to influence demand for vehicles and will make the most sense to include in the input variable. Section 4 below discusses some ways to account both for these less predictable, short run shocks, and the more stable changes discussed above.

2.3.2 Changes to the Generalized Cost of Vehicles

The second main scenario input reflects the fact that changes in new-vehicle policy will influence the average cost and characteristics of new vehicles. This input vector allows forecasts of those changes to be used: the equilibrium model takes those changes and studies how used vehicle prices, and therefore retentions and sales, will react over time. This input vector is labeled *costvec* in the model.

The input reflects the "generalized cost" of a policy, rather than vehicle price only. Generalized cost refers to the change in vehicle price net of the additional benefits or costs to consumers of any simultaneous changes in vehicle characteristics. Greene et al (2018a, 2018b) explores valuation along a range of dimensions in characteristic space. In cases where the characteristics and general desirability of the vehicles remain similar, the generalized cost change is just equal to the price change.

Consider a mandate that 25% of vehicles sold must be electric vehicles (EVs), for example. The case where electric and gasoline vehicles (GVs) are equally desirable (if cost were the same) will be the simplest starting point, and likely a useful benchmark scenario to consider. To the extent information is available on how much less desirable an EV is than a GV (i.e. how much cheaper than the GV it would need to be to get someone to switch) this could also be incorporated. For the simple example, further assume that equivalent EVs will cost $4,000 more to make than existing GVs. In order to sell the required fraction of electric vehicles, avoid selling too many gasoline ones, and break even financially (if we make the simplifying assumption that the market is competitive, so that dealers will go bankrupt if they fail to break even) dealers will need to cross-subsidize their sales.[4] In particular, in this example they would need to solve:

$$
\begin{aligned}
&(1-0.25) * gvtax - 0.25 * evsubs = 0 \\
&evsubs + gvtax = 4{,}000 \\
&\Rightarrow gvtax = 1{,}000
\end{aligned}
$$

In general, under the simplest assumptions as above, the generalized cost of an EV mandate would equal the fraction of EVs required times the added cost of an EV relative to a GV. More detailed measures of generalized cost might additionally include measures of the consumer willingness to pay to avoid EVs (this effectively gets added to the $4,000 in the computation above). Heterogeneity in consumers such that some fraction will purchase EVs anyway, even without the mandate, could also be used to refine the estimate of generalized cost (this will lead to smaller generalized cost because cross-subsidization is only needed for those who wouldn't otherwise buy an EV).

Finally, perceived future fuel and maintenance cost will also enter the generalized cost of a vehicle from the perspective of the buyer. Suppose EVs will save $100 in energy and maintenance cost per year over their life, but the new-vehicle buyer only perceives 4 years of this

[4] This follows the general pattern of mix-shifting to meet constraints, discussed in Jacobsen (2013).

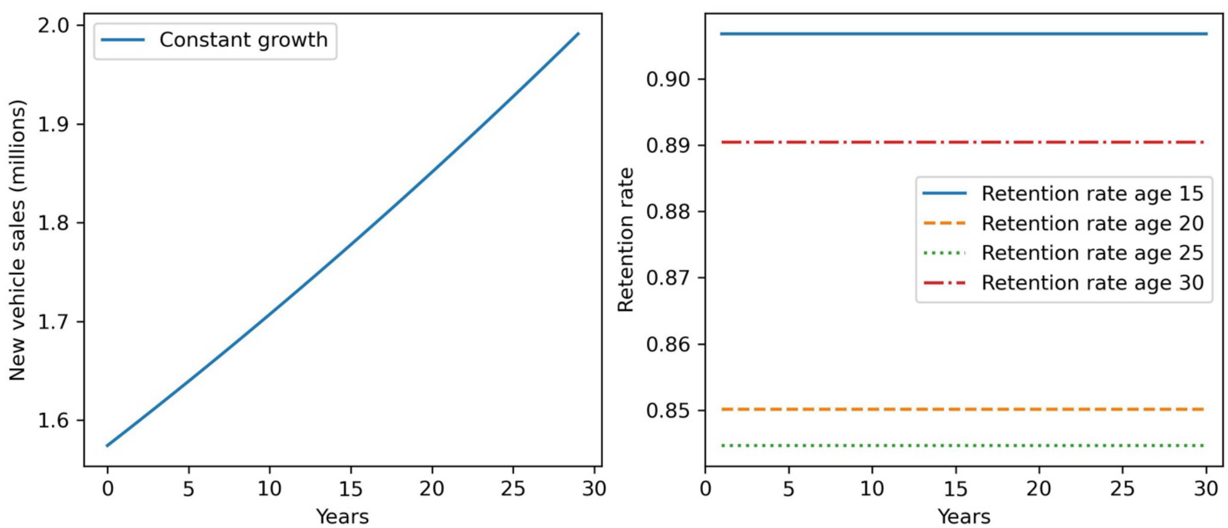
savings. The added cost of an EV (before the implicit tax or subsidy) is now \$3,600 instead of \$4,000, and generalized policy cost falls from \$1,000 to \$900.

SECTION 3.
RESULTS

The DVPop model produces patterns over time in sales and retention rates that are linked together in a dynamic equilibrium. This leads to realistic movements in sales and retention, for example the opposite response of the two in the presence of a generalized cost increase. The time path of sales and retention are produced as output and presented through a series of examples here.

First we examine model output when overall demand grows smoothly and there are no changes to the cost of new vehicles. In other words, the two main scenario inputs are not used. The equilibrium in this case is very simple: retention rates and used vehicle prices are the same year after year and new-vehicle sales grow steadily, at the same historical rate of increase as the rest of the system. The results appear in Figure 3-1.

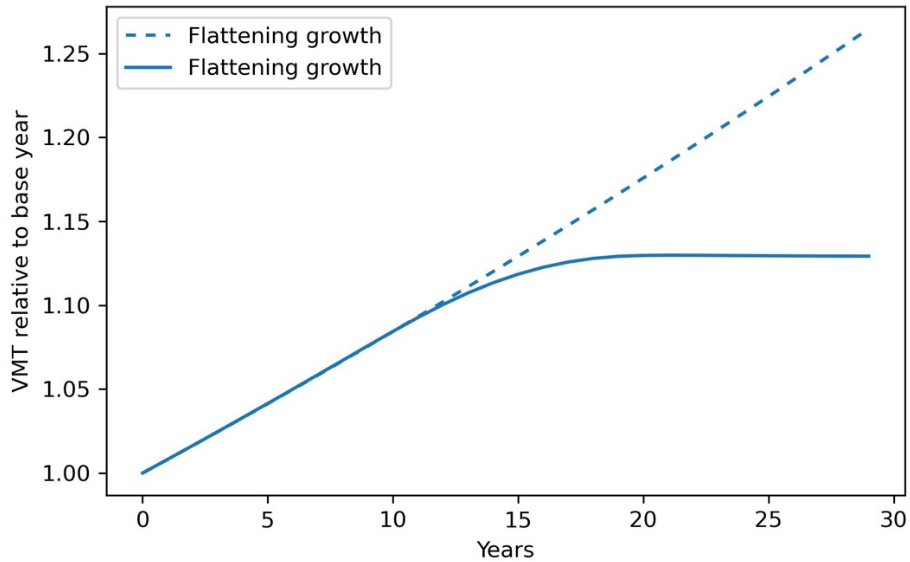
Figure 3-1. Sales and Retention, No Scenario Inputs



3.1 Scenario Analysis: Demand Growth and Policy

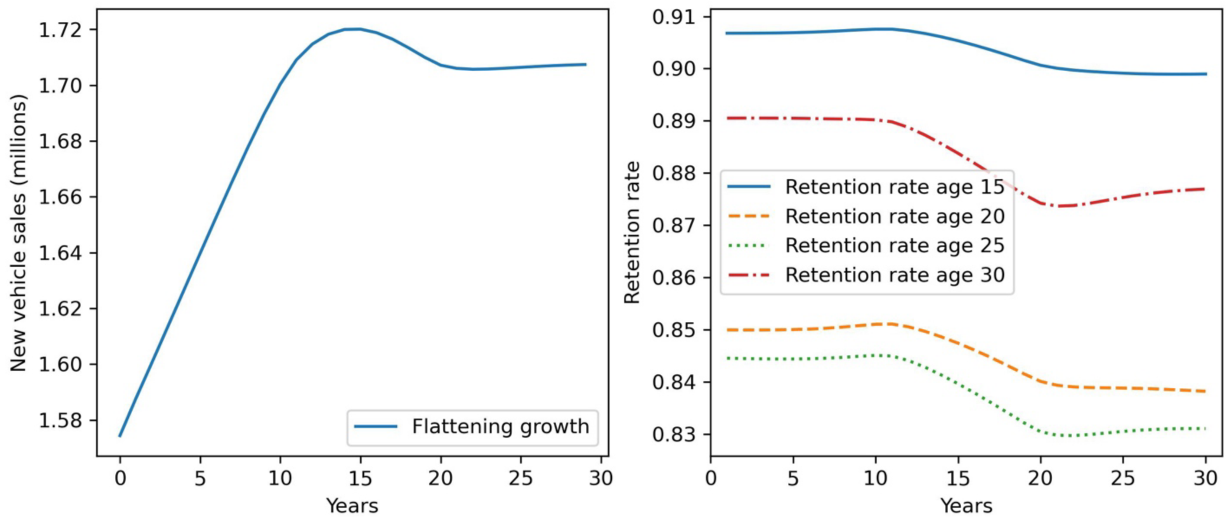
Next consider an example incorporating the aggregate travel demand input. We will set it such that demographic shifts, beginning 10 years from now, slow the travel growth rate down to zero. Note that the VMT flattening is a scenario to demonstrate how the model works. CARB will continue to evaluate how VMT is projected in EMFAC (for example using historical VMT and MPO data). The target path of VMT used in this example is shown in Figure 3-2.

Figure 3-2. Example VMT Growth Path



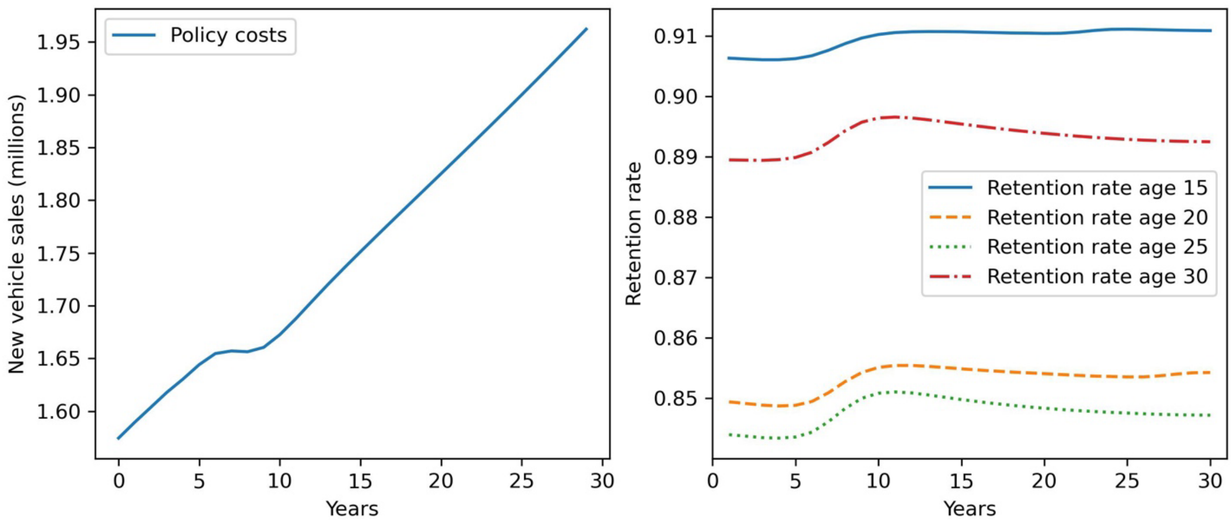
Results for sales and retention rates, when targeting the VMT path above, appear in Figure 3-3. Recall that without incorporating changes in travel demand sales would grow smoothly and retention rates would be constant. The number of used cars therefore also grows smoothly. When accounting for the example travel demand changes, sales flatten around the same time that travel demand growth begins to slow. Then, sales need to decline somewhat in order to keep travel demand flat (in order to counterbalance surpluses in the used market that appear because of sales growth among prior vintages). In the long run, sales then flatten out to just equal the rate needed to replace used vehicles in the new, flat growth, steady state. Retention rates also fall as growth tails off: hitting a low point just as growth reaches zero and then settling. The reason retention rates settle lower than their baseline starting point is that there is no longer pressure on the used car market to help meet a steady sequence of travel demand increases.

Figure 3-3. Sales and Retention, Scenario with Falling Travel Demand



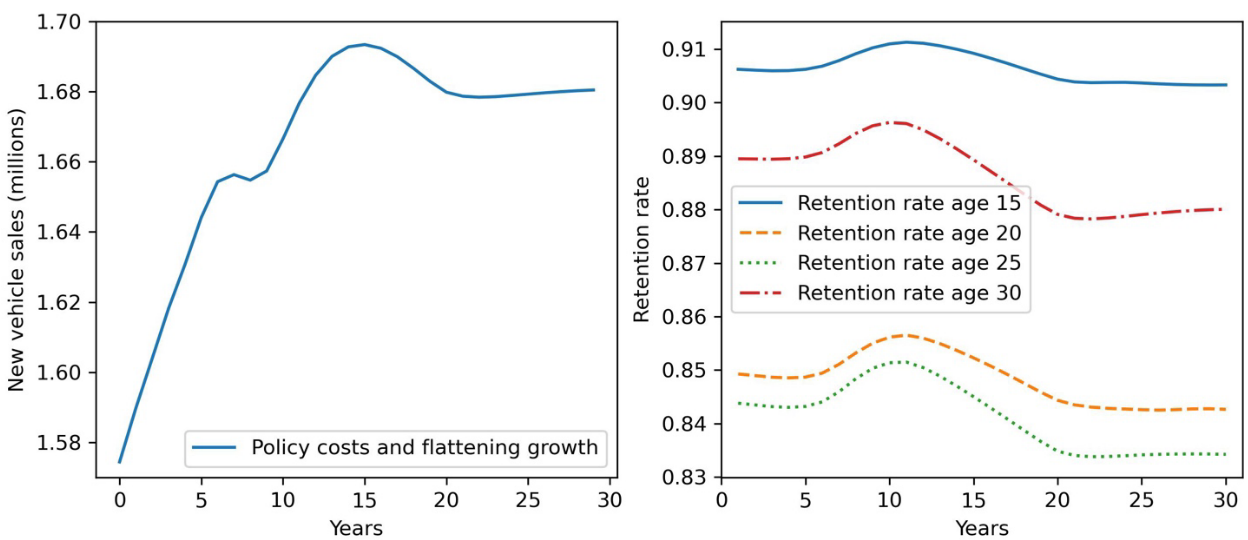
Next we consider an increase to the cost of new vehicles, perhaps as the result of technology mandates in some fraction of new-vehicle sales. This makes use of the second key scenario input: changes in generalized cost. For the moment, we return background growth in travel demand to normal in order to isolate just the effect of the policy cost. The scenario examined here is a \$2,000 increase in cost of the average new vehicle sold, phased in between years 4 and 12. Figure 3-4 shows how the policy curtails vehicle sales abruptly for the first few years after the phase-in begins. Intuitively, vehicle sales can fall fairly dramatically early on since there is a large buffer of relatively new vehicles still available in the used market. Simultaneously, retention rates throughout the used market rise: the model produces this effect endogenously as the result of equilibrium increases in vehicle asset values. After some time in this situation goes by the once-healthy buffer of used vehicles begins to age out and new sales need to resume their growth in order to supply the system. Notice that the growth rate in sales around years 12 and 13 in fact slightly exceeds the background rate: this is the result of accumulated shortages in relatively new vehicles resulting from the early years of the policy.

Figure 3-4. Sales and Retention, Scenario with Policy Costs



Combining the two cases, we now run the full model with both sets of scenario inputs and show the output in Figure 3-5. The slowing background growth in travel demand around the time that sales would have recovered from the policy cost phase-in leads to a much flatter overall path in sales. Retention rates, which had remained elevated in response to the policy cost above, now fall back through more normal levels relatively quickly since the demand slowdown takes hold just as shortages are appearing in the used market. The results here are a full equilibrium and so take into account all interactions between the growth path and policy costs.

Figure 3-5. Sales and Retention, Scenario with Policy Costs and Falling Travel Demand

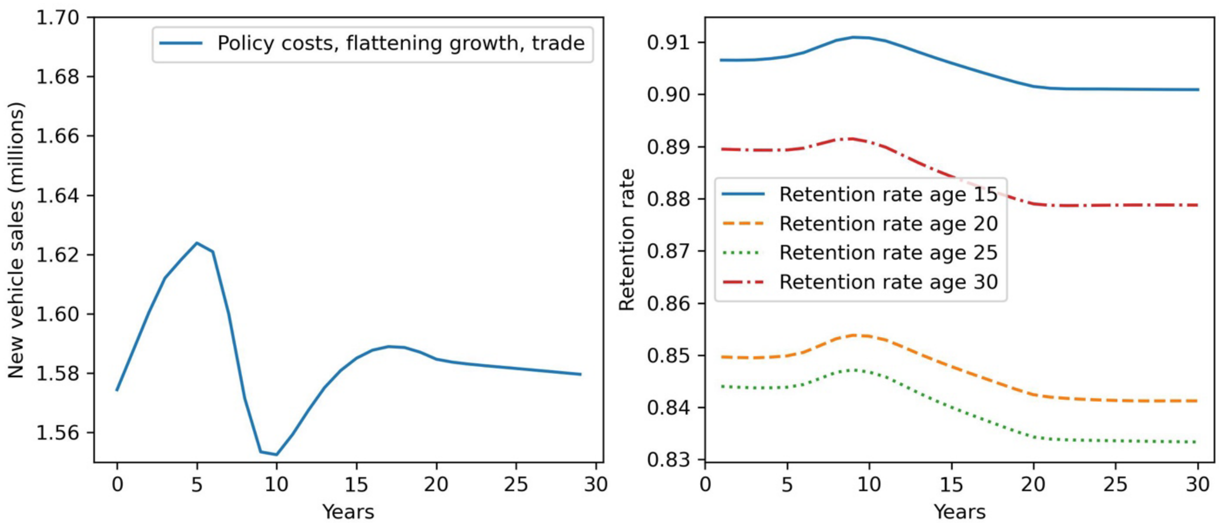


3.2 Scenario Analysis with Trade

Finally, we consider the full model with both sets of inputs in a world where some used vehicle trade to and from California is permitted. The results appear in Figure 3-6 and are quite different, especially in terms of impact on sales, suggesting that accounting for trade may be an important component to consider in future EMFAC modeling efforts. New-vehicle sales now fall much more sharply when the policy cost appears and, unlike the scenarios shown above, they never recover to as high a level. This is because new vehicles are no longer needed as an input to create used vehicles to satisfy the bulk of the vehicle market; used vehicles may be imported directly instead.

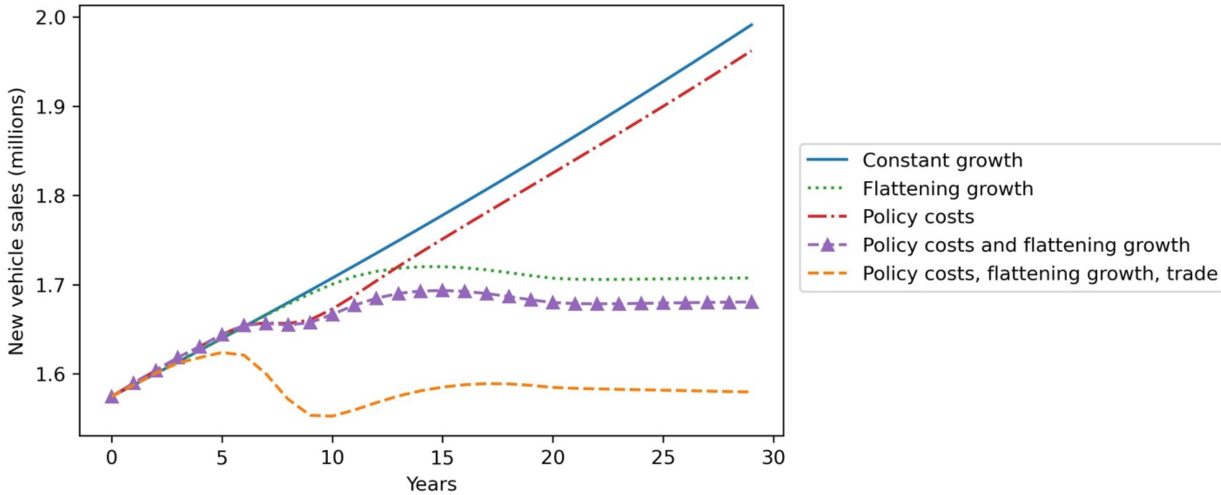
Utilizing the outputs in Figure 3-6 would likely require somewhat more significant changes to EMFAC since now imports with differing emissions, for example in a different composition of electric versus gasoline fuel types, would need to be tracked across ages.

Figure 3-6. Sales and Retention, Scenario with Policy Costs, Falling Travel Demand, and Trade



For a clearer comparison of the projected changes in sales across the scenarios presented above, the sales impacts in Figure 3-1 through Figure 3-6 are combined into a single plot in Figure 3-7 below:

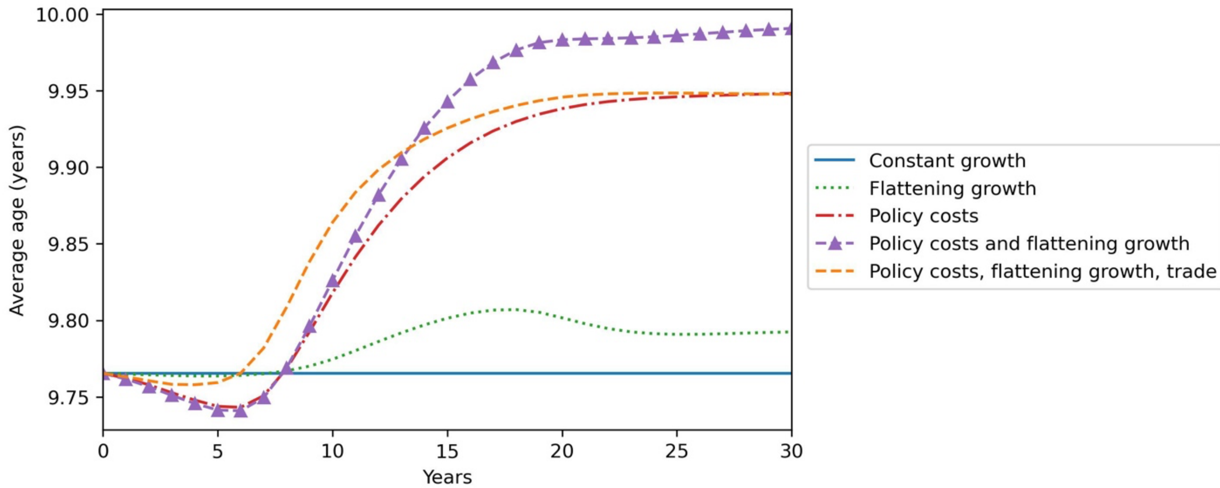
Figure 3-7. Sales Across All Scenarios



3.3 Average Vehicle Age

The changes in sales and retention rates above accumulate over time into changes in the overall age of the fleet, a particularly important variable for emissions measurement given the steep profile of emissions with vehicle age. Figure 3-8 shows the time path of average age for each of the dynamic equilibria shown above. Flattening growth leads to slightly older vehicle ages due simply to the familiar demographic feature that growing populations have younger average ages. The policy costs scenario creates a much larger shift in average age: more expensive new vehicles means that the economy replaces them less often, leading to a higher average age overall. Note that as the policy is phasing in (around years 4-6 for example) average age actually falls slightly. This reflects consumers shifting new vehicle purchases earlier during the policy ramp up in order to avoid the progressively higher costs in each subsequent year. Combining policy and flattening growth further increases average age. Finally, the scenario with trade differs in the speed with which average age increases: vehicle imports allow more rapid response to the increased cost of new vehicles, pushing average age up to its long run level more quickly.

Figure 3-8. Average Vehicle Age Under Alternative Scenarios



SECTION 4.
DISCUSSION

The discussion here provides additional guidance on selecting inputs for scenario analysis, limitations in the model, and the types of analyses that may prove interesting.

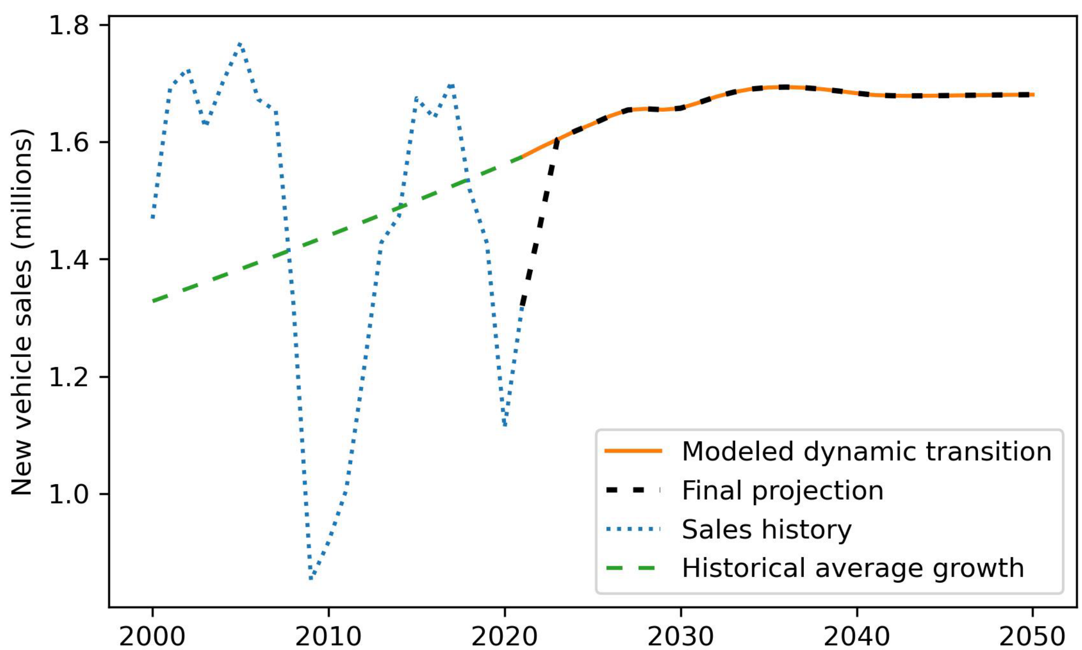
4.1 Choice of the baseline time path of travel demand

The sum of the products of accrual rates and the number of vehicles at each age produces total vehicle miles traveled (VMT) in a given year in the baseline. The growth rate in VMT is an input to the present model. The model then backs out the implied baseline trajectory of vehicle quantities by dividing the input travel demand target by the vector of accruals. This input can be used to include many different sorts of changes CARB may anticipate as possible futures for the private vehicle market, as outlined above.

A limitation of the present model is that the vehicle market in time zero is assumed to be the result of a stable state where retention rates have been fixed (at historical averages) and the growth in sales and the stock have also been fixed through time (at the historical average). The true data in the base year, of course, can be a deviation from the long run history. Sales and/or the total stock could be running higher or lower than the average trajectory in the particular base year selected for the model. To the extent the EMFAC model begins with a year zero that is off the historical path, for example suppose sales are depressed because of a computer chip shortage, modelers may wish to include some time path of a recovery back to normal patterns. This adjustment, from the data back to the long run path, would be in addition to accounting for larger, slower moving, processes like population shifts, housing supply changes, and telecommuting which is done using the main inputs above. We recommend a simple linear approach, with sales returning to their expected trajectory after two years. The sample code provides an example of doing this. Looking at the sales history, the great recession is one case where sales took longer to return to normal: they remained depressed in California for 5 years. In the depths of a similar recession in the future, perhaps a longer-cycle recovery would better be included in the main demand growth projections rather than in the short-term correction outlined here.

Consider the following example based on the trajectory in Figure 3-5 above. Suppose that in year 0 actual vehicle sales had been 5% below the long run average pattern due to a production shock. If we allow 2 years to recover to normal sales levels, a possible forecast path would look like the black dashed line in Figure 4-1 below. The output of DVPop is the solid orange line and starts in the stable state. The dashed black line allows 2 years for sales to transition from observed levels in year 0 back to the modeled path: it could perhaps serve as the best approach to creating an input to EMFAC. A simpler input strategy for EMFAC would be to assume sales "snap" back to normal in year 1. Either way, once sales are recovered to normal levels they would then follow the path modeled here (the solid orange line), which is the dynamic path that accounts for policy and demographic shifts into the more distant future.

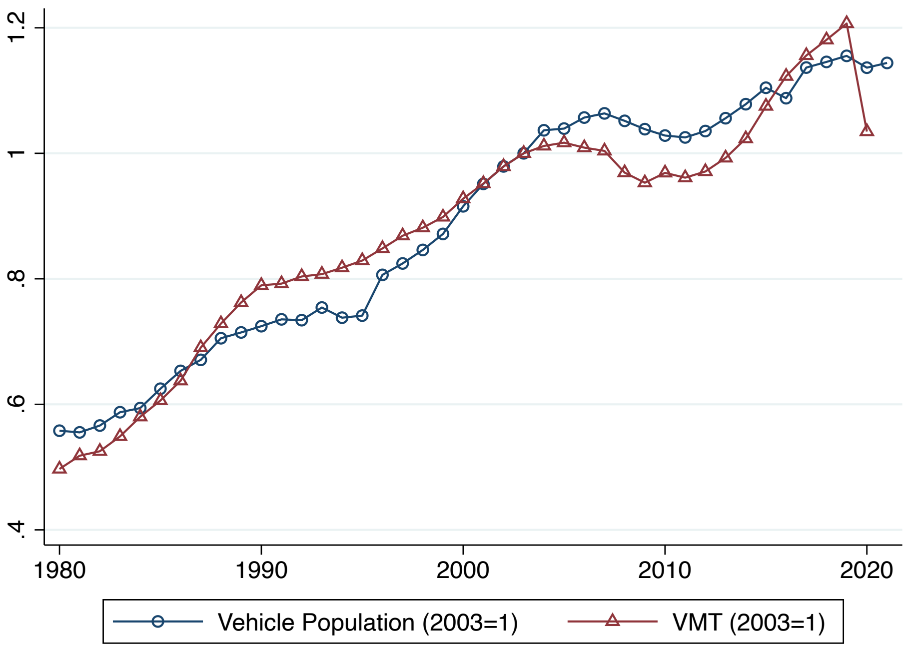
Figure 4-1. Adjustment for Short-Term Sales Shocks Present in the Base Year



4.2 Accruals

If accruals are stable (as assumed to this point), then changes in the demand for private vehicles (the key input in the dynamic model) map directly to changes in travel demand as discussed. Figure 4-2 below considers this assumption and shows that, over long time horizons, the mapping of miles to vehicles has been remarkably stable. Note, however, the sharp disconnect realized in 2020. Vehicle population is basically flat, but aggregate VMT drops sharply, meaning that accruals per vehicle dropped sharply. This was the result of a shock related to the pandemic. More generally, gasoline price and business cycle shocks push accruals up and down in any given year and create the gaps between the two lines in Figure 4-2.

Figure 4-2. Long Run California Vehicle Population and Vehicle Miles Traveled



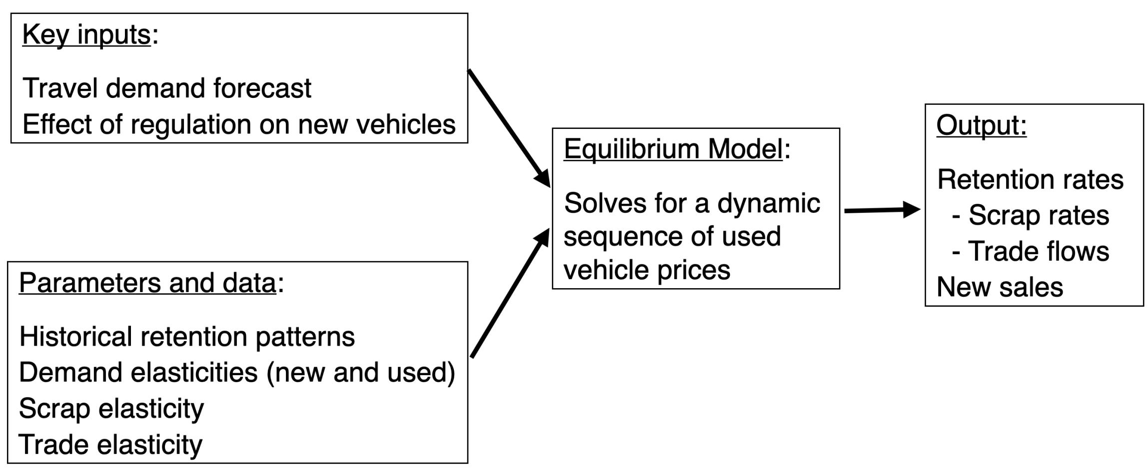
Notes: Data are from CARB and FHWA (https://www.fhwa.dot.gov/policyinformation/statistics.cfm).

To the extent CARB has projections of more permanent changes in accruals (for example, suppose car sharing becomes more popular: accruals per year for each vehicle would trend upwards since each car could be used for multiple commutes) these can be used in DVPop by inputting the variable *vmtprofile* as a matrix (i.e. with dimensions for time and vehicle age) rather than as a vector (with variation only over vehicle age). The model accepts either form. For example, suppose we target 10% greater vmt, using *vmtgrowthvec*, due to anticipated migration into California. At the same time, suppose we think ride sharing will increase accruals per vehicle by 10%. Entering both of these effects as inputs would effectively cancel out in the population model, leading it to project roughly constant demand for vehicles (and therefore roughly constant sales and retentions) overall.

SECTION 5.
SUMMARY AND CONCLUSIONS

The work here develops an economic framework to link together new-vehicle sales, used vehicle retentions, and overall travel demand in California. A calibrated version of the model produces outputs that may be used to add realism and flexibility to the projections of sales and retentions (and therefore the age profile of the vehicle population) in the EMFAC model. Figure 5-1 provides an overview of the model inputs and outputs.

Figure 5-1. Model Inputs and Ouputs



The first key input, shown at top left in the figure, is overall travel demand: how is demand for light-duty vehicle travel in the state forecast to evolve due to factors like demographic shifts, public transportation development, and self-driving technology. The second key input is regulation: how are new regulations likely to influence the overall, or "generalized", cost of new vehicles. These costs should include at least the basic production cost differences needed to comply with regulation net of any subsidy programs. Ideally, changes in fuel cost to the extent they are perceived by new-vehicle buyers, and changes in the valuation of new vehicles due to attributes like size, acceleration, or uncertainties that could make buyers hesitant, can also be included in this summary cost measure.

Other inputs are data and calibrated parameters taken from EMFAC and the economics literature.

The core computation, shown in the middle block, is an equilibrium model that equates vehicle supply and demand at each age and through time. The framework fully accounts for both history (which determines the potential stock at any given point) and for expectations about future vehicle values (which influence the timing of purchase and scrap or retention decisions). Vehicle prices are computed internally and adjust to produce the equilibrium.

The output is a sequence of vehicle populations at each point in time, including new-vehicle sales and used-vehicle retention rates at each age. Sales and retention are modeled simultaneously such that overall travel demand is met in each time period. As vehicles get retired from the fleet, new vehicles must be sold to replace them. This link provides economic consistency and produces intuitive patterns that are not available in current versions of the EMFAC simulation. Finally, the model solution also offers consistency through time in the sense that the aggregate market is forward-looking: more new vehicles are sold (for example using attractive lease rates) when future used vehicle prices (residual values) are expected to be high.

The model also optionally includes a simple representation of trade with the rest of the United States. This is an important area for further investigation, but the current framework allows exploration of a range of scenarios including no-trade, changes in trade flow as a simple function of price differences, and perfect trade. Preliminary examination of the DMV data shows considerable churn of vehicles in and out of the state, such that by age 10 a large fraction of the vehicles registered in California were originally sold in another state. Similarly, data from the rest of the U.S. would likely show that many vehicles originally sold in California actually spend most of their life in other states, for example after being auctioned at the end of a 3-year/30,000 mile lease. If new vehicles of a given vintage are roughly comparable across states the churn and net flow of used vehicles is not all that important for emissions. With current regulation making new vehicles in California much more likely to be zero-emissions than new vehicles in other states, however, vehicle trade patterns become an important determinant of overall pollution inside California. Net retentions in the model output are therefore divided into in-state retentions and those resulting from changes in the net trade flow.

SECTION 6.
RECOMMENDATIONS

The model developed here can be extended along multiple dimensions to provide further detail on changes to the vehicle population, and more realism in accruals. Three extensions that the model may be particularly well-suited to consider are as follows:

I. Disaggregating vehicles at each vintage by fuel type – most simply grouped into "zero emissions" and "conventional" – would allow examination of more subtle changes in retention rates through time. A subsidy to EVs, for example, will draw in buyers who have a used vehicle to trade in, but it may do so differentially depending on if the used vehicle the household currently owns is an EV or not. This in turn will reduce retention rates differentially. This extension would require an additional set of assumptions on the ease with which households are willing to substitute across fuel types by vehicle age, likely the most challenging portion of this type of analysis. The model also becomes more difficult computationally, but a disaggregation to two types probably remains quite feasible.

II. A more detailed trade model could improve understanding of what the used vehicle portion of the California fleet will look like during, and after, the new-vehicle sector switches to zero emissions. The level of churn and net flows of used vehicles at different ages will determine what fraction of vehicles at any given age were originally sold in California versus another state. The first step would be to calibrate the trade frictions in the current version of the model more precisely, choosing values to more realistically represent the ability of vehicles of different ages to move across state lines. The best version of a trade model would include an equilibrium model of two regions and the trade between them: one region would comprise states adopting zero emissions new vehicles with the other states grouped in the second region. Policy questions about fees or age limits to discourage imports from states without zero emissions regulation could be considered.

III. Finally, accruals could be forecast and modeled in detail if data can be gathered on how changes in new vehicle attributes affect driving behavior. At the moment accruals differ by vehicle age following currently available data, but do not change over time. For example, zero emissions vehicle mandates could temporarily discourage accruals if fueling (charging) them proves to be unreliable or involves new patterns that are difficult to learn. Partial self-driving features, on the other hand, could permanently encourage driving if they make it significantly easier or more relaxing to travel in light-duty vehicles.

REFERENCES

Bento, A. M., Goulder, L. H., Jacobsen, M. R., & von Haefen, R. H. (2009). Distributional and Efficiency Impacts of Increased US Gasoline Taxes. *American Economic Review*, *99*(3), 667–699. https://doi.org/10.1257/aer.99.3.667

Berry, S., Levinsohn, J., & Pakes, A. (1995). Automobile Prices in Market Equilibrium. *Econometrica*, *63*(4), 841. https://doi.org/10.2307/2171802

Berry, S., Levinsohn, J., & Pakes, A. (2004). Differentiated Products Demand Systems from a Combination of Micro and Macro Data: The New Car Market. *Journal of Political Economy*, *112*(1), 68–105. https://doi.org/10.1086/379939

Davis, L. W. and M. E. Kahn (2010). International trade in used vehicles: The environmental consequences of NAFTA. *American Economic Journal: Economic Policy* 2 (4), 58–82.

Deaton, A. & Muellbauer, J. (1980) An Almost Ideal Demand System. *American Economic Review*, 70, 312–326.

Dou, X. & Linn, J. (2020). How Do US Passenger Vehicle Fuel Economy Standards Affect New Vehicle Purchases? *Journal of Environmental Economics and Management, 102*, 102332. https://doi.org/10.1016/j.jeem.2020.102332

Goldberg, P. K. (1998). The Effects of the Corporate Average Fuel Efficiency Standards. *The Journal of Industrial Economics, 46*(1), 1–33.

Greene, D., Hossain, A., Hofmann, J., & Beach, R. (2018a). *Consumer Willingness to Pay for Vehicle Attributes: What is the Current State of Knowledge?* EPA-420-R-18-016, Assessment and Standards Division, Office of Transportation and Air Quality, U.S. Environmental Protection Agency, Ann Arbor, MI, July 2018.

Greene, D., Hossain, A., Hofmann, J., Helfand, G., & Beach, R. (2018b). Consumer Willingness to Pay for Vehicle Attributes: What Do We Know? *Transportation Research Part A: Policy and Practice, 118*, 258–279. https://doi.org/10.1016/j.tra.2018.09.013

Gruenspecht, H. K. (1982). Differentiated regulation: The case of auto emissions standards. *American Economic Review* 72(2), 328–31.

Jacobsen, M. R. (2013). Evaluating US fuel economy standards in a model with producer and household heterogeneity. *American Economic Journal: Economic Policy* 5(2), 148–187.

Jacobsen, M., R. Beach, C. Cowell, and J. Fletcher. "The Effects of New-Vehicle Price Changes on New- and Used- Vehicle Markets and Scrappage." U.S. Environmental Protection Agency, Washington, DC. Available from: https://cfpub.epa.gov/si/si_public_record_report.cfm?Lab=OTAQ&dirEntryId= 352754

Jacobsen, M. R., & van Benthem, A. A. (2015). Vehicle Scrappage and Gasoline Policy. *American Economic Review*, *105*(3), 1312–1338. https://doi.org/10.1257/aer.20130935

Jacobsen, Mark, and Sallee, J., Shapiro, J., and van Benthem, A. "Regulating Untaxable Externalities: Are Vehicle Air Pollution Standards Effective and Efficient?" NBER Working paper 30702, December 2022.

Knittel, C. R., & Metaxoglou, K. (2014). Estimation of Random-Coefficient Demand Models: Two Empiricists' Perspective. *Review of Economics and Statistics*, *96*(1), 34–59. https://doi.org/10.1162/REST_a_00394

Leard, B. (2021). *Estimating Consumer Substitution Between New and Used Passenger Vehicles*. RFF Working Paper 19-01 (latest update May 2021). Retrieved from https://media.rff.org/documents/WP_19-01_rev_2021.pdf

Sivak, Michael. (2013). Predicting Vehicle Sales from GDP in 48 Countries: 2005-2011. Retrieved from https://deepblue.lib.umich.edu/bitstream/handle/2027.42/96442/102925.pdf?sequence=1&isAllowed=y

APPENDIX A:
EXAMPLE CALIBRATION TO EMFAC2021 AND ACC II

This appendix re-calibrates the model to an example based on long run VMT growth projections from EMFAC2021 (abstracting from the short-run recovery; this is instead done as in Figure 4-1 above) and a simplified version of projected vehicle costs from ACC II. ACC II cost projections here are for demonstration purposes (a full analysis would require consideration of a range of additional factors). The example here assumes the following:

Generalized costs (the monetized effect of policy on the overall attractiveness of new vehicles in the sense of economic "utility") are assumed to phase in between 2026 and 2035 with the full added cost of new EVs (relative to new gasoline vehicles without ACC II) reaching $1,124 per vehicle in 2035. Note that no adjustment is made for fuel savings in the example here. Different examples could adjust for fuel savings. Also, no adjustment is made for the disutility ("hesitancy") a new vehicle buyer might experience as a result of the mandate. Implicitly, then, the demonstration in this appendix assumes that the fuel savings and disutility from hesitancy (and any other factors) exactly offset such that the overall cost of the policy happens to exactly equal the up-front technology cost.

It may be useful to mention here that binding regulation (binding in the sense that some consumers no longer get their most-preferred vehicle either because its price has gone up or because it is no longer offered in the market) should impose a strictly positive generalized cost: the model here will translate that generalized cost into fewer new vehicle purchases relative to a world without the regulation. This is the case even if the regulation is reducing the financial cost of vehicle ownership. For example, the fuel cost savings in the EV example here could easily mean that EV mandates save people money on net. But, we would still say the mandates have a positive generalized cost to the extent that they prevent some consumers from getting their most-preferred vehicle.

Also note that, by definition, non-binding regulation (for example if more EVs are being purchased than needed to meet a mandate) will have zero generalized cost. Finally, subsidies should be modeled as imposing a negative generalized cost. The model will then show new vehicle sales in the presence of the subsidy as being higher than without it.

Returning to the present example, between 2035 and 2040 we assume the generalized cost of the policy phases out to zero. In 2040 and beyond, the mandated EVs produce exactly the same economic utility as gasoline vehicles would have in the benchmark; new vehicles at that point are no more or less attractive than they would have been without the policy and the ACC II regulation is assumed to stop binding. The figures below show how sales, retention, trade, and vehicle age evolve in the presence of the ACC II mandate when modeled with this approach. A complete analysis of ACC II would require a more nuanced calculation of the generalized cost at different points in the life of the regulation.

Figure A-1. Vehicle Sales History and Projection

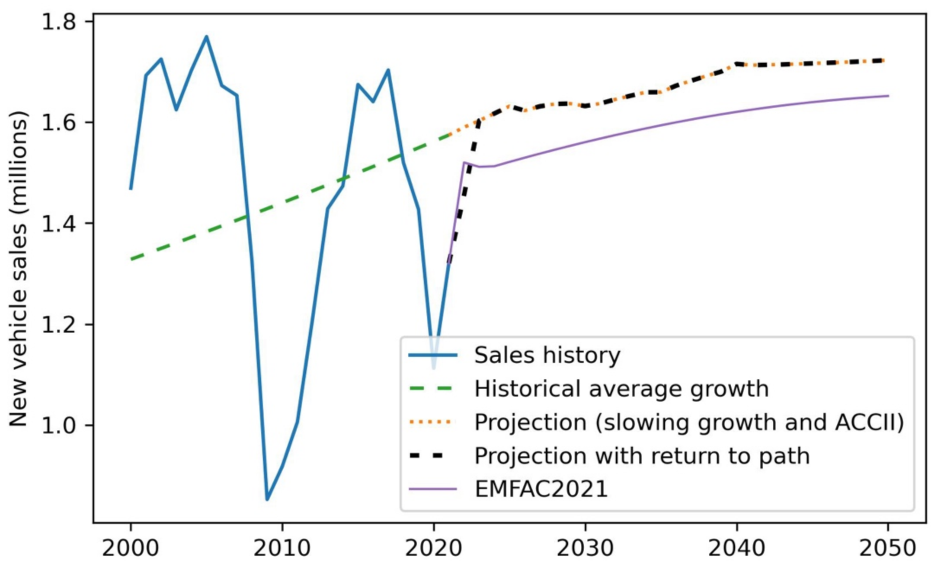


Figure A-2. Vehicle Sales Projections under Alternative Scenarios

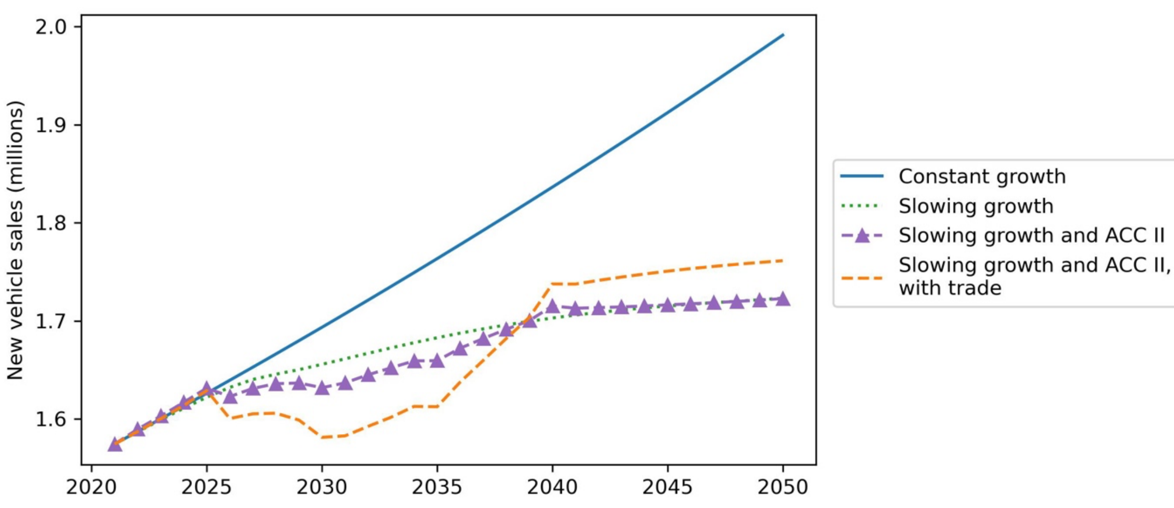


Figure A-3. Effects of Trade on Retention, Sample of Ages 3 and 20

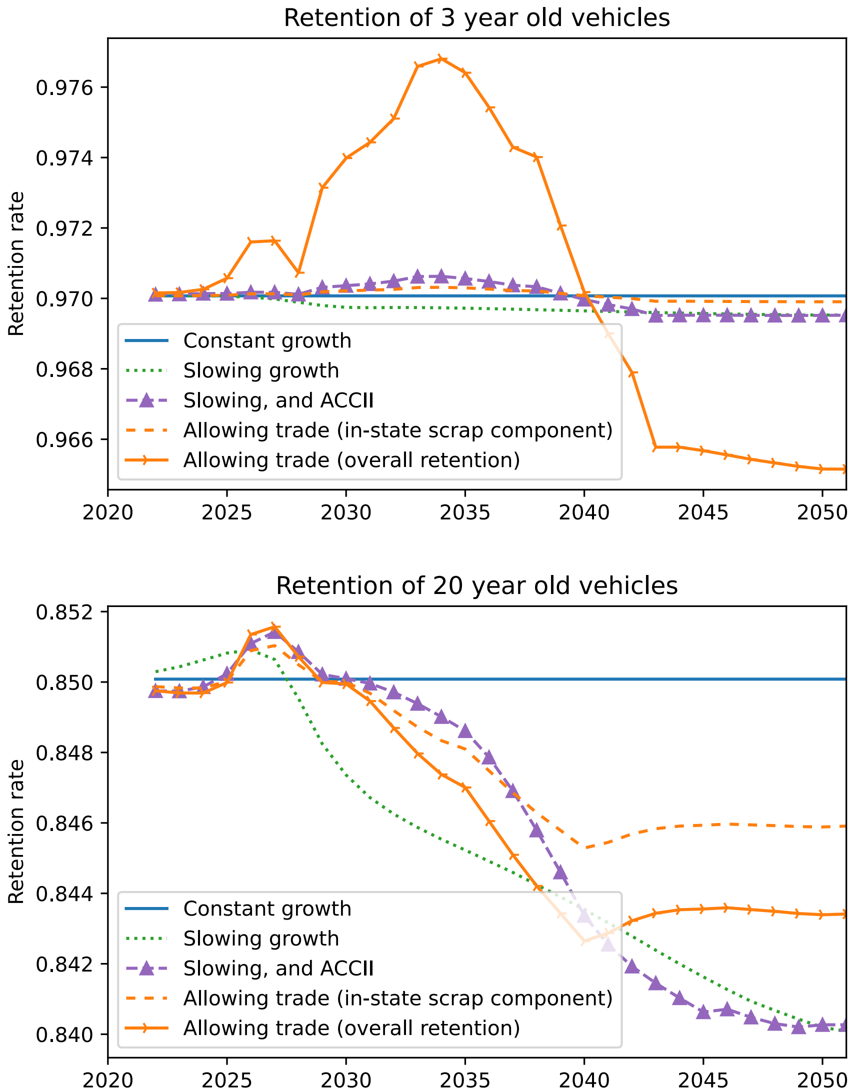
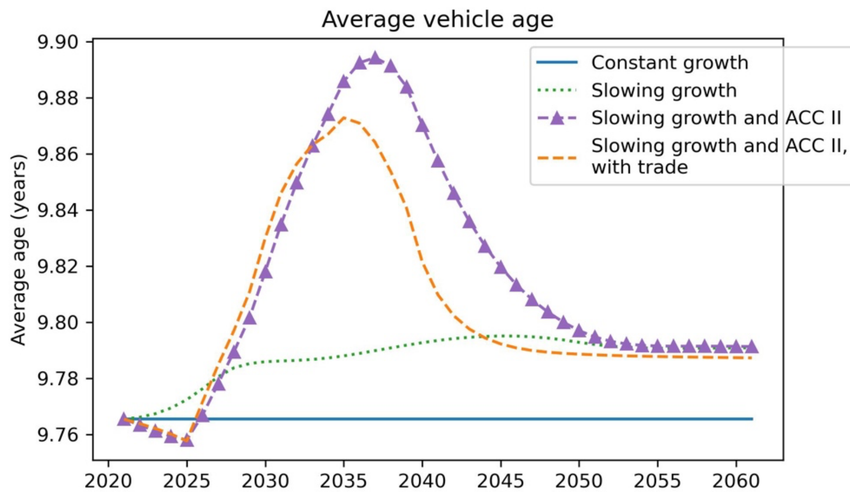


Figure A-4. Average Vehicle Age



APPENDIX B:
MODEL YEAR AND AGE ACCOUNTING

Accurate modeling of emissions requires tracking the model year of a vehicle, as this is the primary indicator of the emissions control systems and regulation that is applicable to any given vehicle in the population. However, the calendar age of a vehicle (i.e. number of months since it was driven off a dealer's lot) is also a relevant quantity, most notably in the reporting of vehicle sales which tends to abstract from model years and instead focuses on units sold in a calendar month or year.

Modeling vehicle demand is also made more intuitive by a calendar age approach, and so the system here needs to be able to translate back and forth between "model year" accounting systems (as in EMFAC) and a "calendar age" system.

This is handled by two helper routines in the code ("q_to_myq", and "myq_to_q") that shift demand by 11% (the typical fraction of vehicles of the previous model year that were purchased less than 12 months before the cut of DMV data used in EMFAC) through the whole population. Retention rates may be input using either definition and results returned either way. A sample is shown in the figures below to demonstrate the typical differences in patterns depending on accounting definition. Importantly, the total number of vehicles in the fleet is identical regardless of accounting, this just amounts to relabeling of ages.

Figure B-5. Vehicle Population Using Alternative Definitions of Age

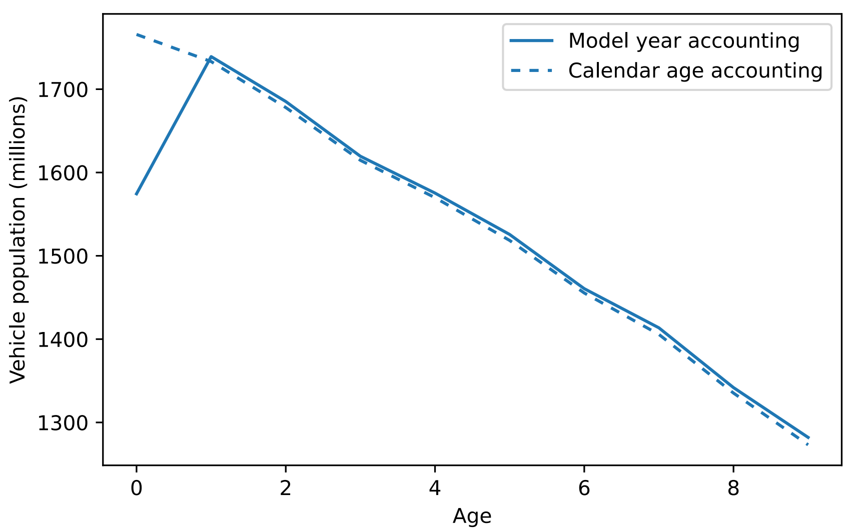
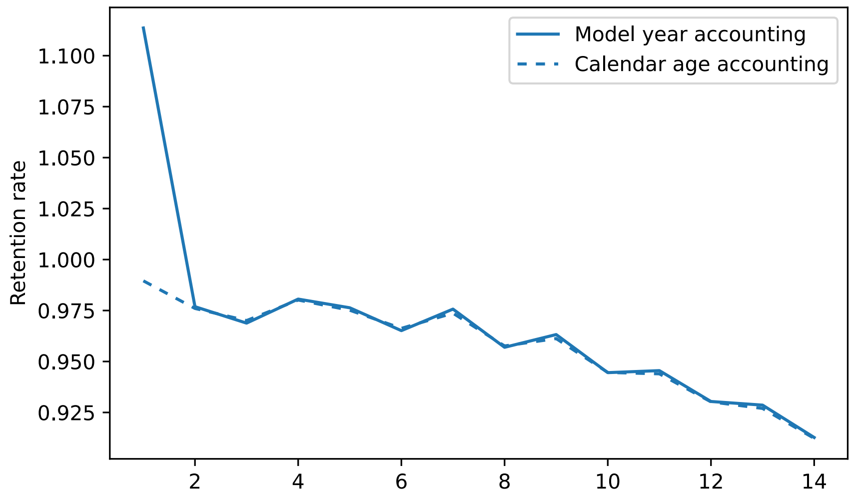


Figure B-6. Retention Rates Using Alternative Definitions of Age



APPENDIX C:
MODEL CODE

```
#
# Dynamic Vehicle Population Model (dvpop)
# Version: 1/31/23
#
# Mark Jacobsen
# University of California, San Diego
#
# Based on work in:
# Jacobsen, M, Beach, R, Cowell, C and Fletcher, J. "The Effects of New-Vehicle Price Changes on New-
# and Used-Vehicle Markets and Scrappage." Washington DC: U.S. Environmental Protection Agency, 2021.
#

import numpy as np
from scipy.optimize import fsolve, minimize
import warnings
import copy


class LREqm:
    # variables to store a long run, steady-state equilibrium
    def __init__(self, nc):
        self.nc = int(nc)
        self.ng = int(nc+1) # number of goods in an equilibrium is number of cars + 1
        # components of an equilibrium
        self.q = np.zeros(self.ng, dtype=float) # quantities
        self.r = np.zeros(self.ng, dtype=float) # per-period prices
        self.p = np.zeros(self.nc, dtype=float) # asset prices
        self.s = np.zeros(self.nc, dtype=float) # scrap rates
        self.h = np.zeros(self.nc, dtype=float) # expected repair spending
        self.i = np.zeros(self.nc, dtype=float) # imports
    @property
    def myq(self):
        myq = np.zeros(self.nc, dtype=float)
        myq = q_to_myq(self.q[:], self.nc)
        return myq


class Eqm:
    # variables to store a dynamic equilibrium with nt time periods
    def __init__(self, nc, nt):
        self.nc = int(nc)
        self.ng = int(nc+1) # number of goods in an equilibrium is number of cars + 1
        self.nt = int(nt)
        # components of an equilibrium
        self.q = np.zeros((self.ng,self.nt), dtype=float) # quantities
        self.r = np.zeros((self.ng,self.nt), dtype=float) # per-period prices
        self.p = np.zeros((self.nc,self.nt), dtype=float) # asset prices
        self.s = np.zeros((self.nc,self.nt), dtype=float) # scrap rates
        self.h = np.zeros((self.nc,self.nt), dtype=float) # expected repair spending
        self.i = np.zeros((self.nc,self.nt), dtype=float) # imports
    def tfill(self, lre, M, mint=0, maxt=None):
        # fill time periods (up to maxt) with the LREqm in lre (where q grows with growth in M)
        if maxt==None: maxt = self.nt
        for j in range(self.ng):
            self.q[j,mint:maxt] = lre.q[j] * (M[mint:maxt]/M[0])
            self.r[j,mint:maxt] = lre.r[j]
        for j in range(self.nc):
            self.p[j,mint:maxt] = lre.p[j]
            self.s[j,mint:maxt] = lre.s[j]
            self.h[j,mint:maxt] = lre.h[j]
            self.i[j,mint:maxt] = lre.i[j] * (M[mint:maxt]/M[0])
    @property
    def nr(self):
        nrmat = np.zeros((self.nc,self.nt), dtype=float)
```

```
        for j in range(1,self.nc):
            nrmat[j,1:self.nt] = self.q[j,1:self.nt] / self.q[j-1,0:self.nt-1]
        return nrmat
    @property
    def myq(self):
        myq = np.zeros((self.nc,self.nt), dtype=float)
        for t in range(self.nt):
            myq[:,t] = q_to_myq(self.q[:,t], self.nc)
        return myq
    @property
    def mynr(self):
        mynr = np.zeros((self.nc,self.nt), dtype=float)
        for j in range(1,self.nc):
            mynr[j,1:self.nt] = self.myq[j,1:self.nt] / self.myq[j-1,0:self.nt-1]
        return mynr


class Scenario:
    def __init__(self, nc, nt, br, totalstock0, ll, bp, gg, eemode, ee_input, retentionmy=False,
            vmtprofile=0, eefalloff=0.08, eerelativeused=1.0, dd=0.03, trade=False, tradeslope=None):

        # inputs for elasticity calibration
        self.eemode = eemode
            # newoutside: takes two elasticities as inputs, new vehicle demand, and total vehicle demand (wrt outside good)
            # precalibrated: reads a pre-calibrated matrix of demand derivatives in directly
        self.eefalloff = eefalloff # target rate at which substitution elasticities fall off with age differences
        self.eerelativeused = eerelativeused # own-price elasticity of oldest used vehicles relative to new vehicles

        # toggles, counters
        self.itco = 0
        self.lastsumobj = 0

        # model inputs
        self.nc = int(nc) # number of cars
        self.ng = int(nc+1)  # number of goods
        self.nt = int(nt) # number of time periods
        self.dd = np.double(dd) # discount rate (delta)
        self.trade = trade # if vehicle import/export is allowed
        self.gg = np.double(gg) # scrap elasticity (gamma)
        self.ll = np.double(ll) # background growth rate
        self.totalstock0 = np.double(totalstock0) # baseline scale of the total vehicle stock
        self.aa = np.zeros(self.nc, dtype=float) # scale parameters in scrap fn

        # 3 equivalent matrices for expressing the demand system:
        self.tt = np.zeros((self.ng,self.ng), dtype=float) # matrix of theta parameters for the demand system
        self.ee = np.zeros((self.ng,self.ng), dtype=float) # matrix of own and cross-price elasticities (at baseline) for the demand system
        self.dqdr = np.zeros((self.ng,self.ng), dtype=float) # matrix of dqdr values (at baseline) for the demand system

        self.bb = np.zeros(self.ng, dtype=float) # demand system scale parameters (calibrated to match bq)
        self.M = np.zeros(self.nt, dtype=float) # exogenous time path of income
        self.b = LREqm(self.nc) # create a baseline equilibrium (t=0)
        self.b.s = (1-br) # convert retention rates to scrap rates
        if retentionmy:
            # read in retention is by model year, convert to age-based accounting first:
            tempq = np.zeros(self.nc, dtype=float)
            tempq[0] = 1
            for j in range(1,self.nc): tempq[j] = br[j] * tempq[j-1]
            tempq = myq_to_q(tempq,self.nc)
            for j in range(1,self.nc):
                self.b.s[j] = 1 - tempq[j] / tempq[j-1]
        self.b.p = np.double(bp) # baseline car prices read in
        if np.shape(vmtprofile):
```

```
    # repeat final column if fewer than nt time periods have been read in
    self.vmtprofile = np.hstack((vmtprofile, np.tile(vmtprofile[:, [-1]], nt -
    vmtprofile.shape[1])))

# use historical retention and growth rates to construct baseline steady state stock
self.b.q[0] = 1
for j in range(1,self.nc):
    self.b.q[j] = (1 - self.b.s[j]) * self.b.q[j-1] / (1+self.ll)
temp = np.sum(self.b.q[0:nc])
self.b.q[0:nc] = totalstock0 * self.b.q[0:nc] / temp
self.b.q[self.ng-1] = 1 # placeholder for outside good

# implied scale parameter in the scrap function (alpha)
self.aa = self.b.s / (np.float_power(self.b.p,self.gg))
# convert baseline purchase prices to series of implied rental prices
self.b.r, self.b.s, self.b.h = self.p_to_rsh(self.b.p)

# calibrate baseline level of outside good using outside good share ogs
# note: income effects are assumed to be neutral in current demand system so any value for ogs
    (>0 and <1) produces same output
self.ogs = 0.95
self.b.q[self.ng-1] = self.ogs / (1-self.ogs) * np.sum(self.b.q[:self.nc] * self.b.r[:self.nc])

# baseline income growing at rate ll
self.M[0] = np.sum(self.b.q * self.b.r) # baseline income in time 0
for t in range(1, self.nt):
    self.M[t] = (1+self.ll) * self.M[t-1]

# baseline time path (steady state)
self.bt = Eqm(self.nc, self.nt)
self.bt.tfill(self.b, self.M)

# holder for transition path to be estimated
self.polt = Eqm(self.nc, self.nt)
self.polt.tfill(self.b, self.M)

# trade model
if tradeslope is None:
    self.tradeslope = np.zeros(self.nc, dtype=float)
else: self.tradeslope = tradeslope

# demand system substitution parameters
if eemode=="precalibrated":
    for i in range(self.ng):
        for j in range(self.ng):
            self.set_dqdr(i,j,ee_input[i,j]) # inputs are the full matrix of derivatives, copy
            it into the scenario
else:
    self.eecalib_targets = ee_input # inputs are the calibration targets

if eemode=="newoutside":

    # the diagonal is determined by elasticity assumptions, fill that first
    # get depreciation component for producing elasticities independent of repair spending:
    self.r0_deponly = self.b.r.copy()
    for j in range(self.nc-1):  # (for the oldest car we already have depreciation since car is
    no longer repaired)
        self.r0_deponly[j] = self.b.p[j] - (1 - self.b.s[j+1])*self.b.p[j+1]/(1 + self.dd)
    # set diagonal
    for j in range(self.nc):
        self.set_dqdr(j,j, self.eecalib_targets[0] * (1 + j/(self.nc-1)*(self.eerelativeused-1))
    * self.b.q[j] / self.r0_deponly[j])

    # get starting values for remaining cells (idx values will be needed to exactly complete the
    matrix)
    idx = 0
    for j in range(self.nc-3):
        for i in range(j+2,self.nc):
            idx = idx + 1
```

```
        startvals = np.zeros(idx, dtype = float)

        minout = minimize(self.calibeeobj, startvals)
        self.calibeeobj(minout.x, verbose=True)

        # set parameters
        for i in range(self.nc):
            for j in range(self.nc):
                self.set_dqdr(i,j,1.0*self.dqdr[i,j])

        # solve for outside good
        for j in range(self.nc):
            self.set_tt(self.ng-1,j, -1.0*np.sum(self.tt[:self.ng-1,j]))
            self.set_tt(j,self.ng-1, -1.0*np.sum(self.tt[:self.ng-1,j])) # symmetry
        self.set_tt(self.ng-1,self.ng-1, -1.0*np.sum(self.tt[:self.ng-1,self.ng-1])) # adding

        # calibrate the beta parameters (scale demand system to match baseline)
        for j in range(self.ng):
            self.bb[j] = (self.b.q[j] * self.b.r[j]) / self.M[0] - np.sum([self.tt[j,k] *
        np.log(self.b.r[k]) for k in range(self.ng)])

    # np.savetxt("theta.csv", self.tt, delimiter=",")

    # calibrate the beta parameters to baseline quantity
    for j in range(self.ng):
        self.bb[j] = (self.b.q[j] * self.b.r[j]) / self.M[0] - np.sum([self.tt[j,k] *
        np.log(self.b.r[k]) for k in range(self.ng)])


def sh(self, p, aa, gg):
    # constant elasticity of scrappage, takes price p and parameters and returns scrap rate s and
        expected repair cost h
    s = aa * (np.float_power(p,gg))
    if s >= 1: # in case the solver is exploring an area where the scrap rate exceeds 1.  This will
        never be an equilibrium point but the repair cost function should still return a value
        h = p
    else:
        if gg == 0.00 or aa == 0.00:
            h = 0
        elif gg == -1.00:
            h = aa * p * (np.log(aa)-np.log(p)) / (aa-p)
        else:
            h = (-gg * np.float_power(aa,(-1/gg)) + aa * gg * np.float_power(p,(1+gg))) /
        ((1+gg)*(aa * np.float_power(p,gg) - 1))
    return s, h


def dem(self, M, r):
    # main demand system, returns q for any M and r
    q = np.empty(self.ng, dtype=float)
    for j in range(self.ng):
        q[j] = (M/r[j]) * (self.bb[j] + np.sum([self.tt[j,k] * np.log(r[k]) for k in
        range(self.ng)]))
    return q


def p_to_rsh(self, p):
    # convert a vector of purchase prices to implied rental prices, scrap rates, and repair costs
    r = np.empty(self.ng, dtype=float)
    s = np.empty(self.nc, dtype=float)
    h = np.empty(self.nc, dtype=float)
    for j in reversed(range(self.nc)):
        if j > 0: s[j], h[j] = self.sh(p[j],self.aa[j],self.gg)
        if j == self.nc-1:   # oldest cars
            r[j] = p[j]
        else:
            r[j] = p[j] - (1 - s[j+1])*(p[j+1]-h[j+1])/(1 + self.dd)

    h[0] = 0
```

```
    r[self.ng-1] = 1 # normalize price of outside good to 1
    return r,s,h


def exdemand1(self, p1, costadd):
    # takes a generalized cost increase (costadd) and a guess for p1 (vector of used car prices) and
        returns excess demand for used cars, i.e. qD1 - qS1
    self.pol.p[0] = self.b.p[0] + costadd
    self.pol.p[1:] = p1

    self.pol.r,self.pol.s,self.pol.h = self.p_to_rsh(self.pol.p)
    self.pol.q = self.dem(self.M[0], self.pol.r)

    qS = np.zeros(self.nc, dtype=float)
    qS[0] = self.pol.q[0]
    for j in range(1,self.nc):
         qS[j] = (1-self.pol.s[j]) * (self.pol.q[j-1] / (1+self.ll))

    if self.trade:
        self.pol.i[1:self.nc] = (self.pol.p[1:self.nc] - self.b.p[1:self.nc]) *
        self.tradeslope[1:self.nc] * self.b.q[1:self.nc]

    return self.pol.q[1:self.nc] - qS[1:] - self.pol.i[1:]


def steadystate(self, costadd):
    # solve for the steady state when generalized cost of "costadd" is added to p0
    self.pol = LREqm(self.nc) # create a policy equilibrium
    # starting values for used cars
    startval = self.b.p[1:] * (1+costadd/self.b.p[0])

    fsolveoutput = fsolve(self.exdemand1, startval, costadd)
    solnexdem = self.exdemand1(fsolveoutput, costadd)

    # create a counterfactual steady state policy time path for reference
    self.sspolt = Eqm(self.nc, self.nt)
    self.sspolt.tfill(self.pol, self.M)

    return self.pol


def dynamicvmttarget(self, vmtgrowthvec, costaddvec, tolerance=0.1, iterate_factor=0.8, max_try=100,
        printprogress=False):
    # solves dynamic transition path that targets a particular VMT growth path (vmtgrowthvec) and
        includes policy-related changes to new vehicle costs (costaddvec)
    # VMT growth rate assumed to remain at growth rate at len(vmtgrowthvec) if the vector is shorter
        than nt
    # this method updates the self.M vector (aggregate spending on travel) such that it leads to the
        target VMT trajectory

    # starting guess for the calibrated M vector (growth in spending at the rate of desired VMT
        growth)
    for t in range(1, len(vmtgrowthvec)+1):
        self.M[t] = (1+vmtgrowthvec[t-1]) * self.M[t-1]
    for t in range(len(vmtgrowthvec)+1, self.nt):
        self.M[t] = (1+vmtgrowthvec[-1]) * self.M[t-1]

    # generate sequence of VMT targets from the input sequence of growth rates
    vmtseq = np.zeros(self.nt, dtype=float) # sequence of VMT targets
    vmtseq = self.M / self.M[0] * np.sum(self.polt.q[0:self.nc,0] * self.vmtprofile[:,0])
    #print('\nVMT target growth sequence: ', vmtseq[0:]/vmtseq[0])

    vmtcalibratedbaseline = self.dynamicmodel(costaddvec, tolerance=tolerance,
        iterate_factor=iterate_factor, max_try=max_try, printprogress=printprogress,
        longoutput=False, startmyopic=1)
    if printprogress==True: print('entering main VMT target loop')
    # solve the complete dynamic equilibrium up to max_xtry more times, updating the M vector each
        time to more closely target VMT:
```

```
        max_xtry = 8
        for xtry in range(max_xtry):
            currvmt = [sum(self.polt.q[0:self.nc,t] * self.vmtprofile[:,t]) for t in range(self.nt)]
            #print('\nCurrent VMT sequence', currvmt)
            vmtratio = vmtseq/currvmt
            if printprogress==True: print('\nVMT tol ',xtry,max(np.abs(vmtratio[:self.nt-1]-1)))
            if max(np.abs(vmtratio[:self.nt-1]-1)) < 0.001:
                print('Matched VMT trajectory')
                break
            self.M = vmtratio * self.M

            vmtcalibratedbaseline = self.dynamicmodel(costaddvec, tolerance=tolerance,
            iterate_factor=iterate_factor, max_try=max_try, printprogress=printprogress,
            longoutput=False, startmyopic=1)

        if xtry == max_xtry - 1:
            print('Warning: VMT target tolerance may not be met')

        return vmtcalibratedbaseline


    def dynamicmodel(self, costaddvec, announcet=1, tolerance=0.1, iterate_factor=0.8, max_try=100,
            printprogress=False, longoutput=False, startmyopic=1):
        # solves dynamic transition path from announcet to t = T-1 (steady state expectations begin at
            T-1)
        # income path is taken as given in M
        # net effects of new-car policy are in vector costaddvec
        # if announcet > 1, polt with no policy is used to that point

        # if we are permanently changing new vehicle costs, solve for new LR steady state:
        if costaddvec[self.nt-1] > 0:
            lrpol = self.steadystate(costaddvec[self.nt-1])
            self.polt.tfill(lrpol, self.M, mint=announcet)

        Ep = np.zeros((self.nc, self.nt), dtype=float) # expected price vector
        if printprogress:
            print('Dynamic model iteration: ', end="", flush=True)

        if startmyopic >= 1:
            # solve time sequence one time through using myopic expectations (starting values)
            sumexdem = 0
            for t in range(announcet,self.nt):
                startval = self.polt.p[1:,t-1]
                fsolveoutput = fsolve(self.exdemandt, startval, args=(Ep[:,t], t, costaddvec[t],
            "updatingmyopic"))
                exdemtemp = self.exdemandt(fsolveoutput, Ep[:,t], t, costaddvec[t], "updatingmyopic") #
            run once at solution to update
                sumexdem += np.sum(np.abs(exdemtemp))
            print('myop1 sumexdem',sumexdem)

        saveits = [copy.deepcopy(self.polt)] # save starting iteration, use as startvals for exdem
            solver below

        # iterate on price expectations max_try times, moving iterate_factor each time
        for iter in range(max_try):
            if printprogress: print(iter, end=" ", flush=True)
            lastp = copy.deepcopy(self.polt.p)
            if iter==0:
                Ep = lastp
            else:
                Ep = Ep + iterate_factor * (lastp - Ep)

            sumexdem = 0
            for t in range(announcet,self.nt-1):
                startval = 1.0*saveits[iter].p[1:,t]  # starting values at solution to last iteration

                with warnings.catch_warnings():
                    warnings.simplefilter("ignore") # don't print fsolve warnings (instead track
            numerically using sumexdem)
```

```
                fsolveoutput = fsolve(self.exdemandt, startval, args=(Ep[:,t+1], t, costaddvec[t],
        "standard"))

            exdemtemp = self.exdemandt(fsolveoutput,Ep[:,t+1], t, costaddvec[t], "standard")
            sumexdem += np.sum(np.abs(exdemtemp))

        # expectations are in steady state in final year so use directly:
        t = self.nt-1
        startval = saveits[iter].p[1:,t]  # starting values at solution to last iteration
        fsolveoutput = fsolve(self.exdemandt, startval, args=(Ep[:,t], t, costaddvec[t],
        "updatingmyopic"))
        exdemtemp = self.exdemandt(fsolveoutput,Ep[:,t], t, costaddvec[t], "updatingmyopic")
        sumexdem += np.sum(np.abs(exdemtemp))

        saveits.append(copy.deepcopy(self.polt))

        convexpect = max(sum(np.abs(Ep-self.polt.p))) # sum of differences between actual and
        expected prices in time period that is farthest from converging
        if printprogress:
            print('('+str(round(convexpect,4))+', '+str(round(sumexdem,4))+') ', end="", flush=True)
        if convexpect < tolerance:
            break

    # warnings and convergence checks
    if convexpect > tolerance:
        print('\nWarning: tolerance not met in convergence of price expectations')
    elif printprogress:
        print('\nSuccessful convergence of expectations, sum of differences: ', round(convexpect,4))
    if sumexdem > 0.1:
        print('Warning: dynamic model sumexdem exceeds 0.1, sum of excess demand: ',
         round(sumexdem/self.nt,9))
    elif printprogress:
        print('Successful convergence of used market, sum of excess demand: ',
         round(sumexdem/self.nt,9))

    if longoutput:
        return copy.deepcopy(self.polt), self.bt, self.sspolt, saveits, Ep
    else:
        return copy.deepcopy(self.polt)


def pEp_to_rsh(self, p, Ep):
    # convert a vector of purchase prices and expected prices next year to rental prices, scrap
        rates, and repair costs
    r = np.empty(self.ng, dtype=float)
    s = np.empty(self.nc, dtype=float)
    h = np.empty(self.nc, dtype=float)
    Es = np.empty(self.nc, dtype=float)
    Eh = np.empty(self.nc, dtype=float)

    for j in reversed(range(self.nc)):
        if j>0: s[j], h[j] = self.sh(p[j],self.aa[j],self.gg)
        if j == self.nc-1:   # oldest cars
            r[j] = p[j]
        else:
            Es[j+1], Eh[j+1] = self.sh(Ep[j+1],self.aa[j+1],self.gg)
            r[j] = p[j] - (1 - Es[j+1])*(Ep[j+1]-Eh[j+1])/(1 + self.dd)

    h[0] = 0

    r[self.ng-1] = 1 # normalize price of outside good to 1
    return r,s,h


def exdemandt(self, p1, Ep, t, costadd, exp_mode):
    # takes values for p1 (vector of used car prices), Ep (expected vector of all car prices next
        year) and returns excess demand for used cars, i.e. qD1 - qS1 in time t.  costadd is the
        generalized cost increase for price of new cars in time t
    self.polt.p[0,t] = self.bt.p[0,t] + costadd
```

```
        self.polt.p[1:,t] = p1

        if (exp_mode=="updatingmyopic"):
            Ep = self.polt.p[:,t]

        elif (exp_mode=="updatingpartial"):
            Ep = 0.8*self.polt.p[:,t] + 0.2*Ep

        self.polt.r[:,t],self.polt.s[:,t],self.polt.h[:,t] = self.pEp_to_rsh(self.polt.p[:,t], Ep)
        self.polt.q[:,t] = self.dem(self.M[t], self.polt.r[:,t])

        if self.trade:
            self.polt.i[1:self.nc,t] = (self.polt.p[1:self.nc,t] - self.bt.p[1:self.nc,t]) *
            self.tradeslope[1:self.nc] * self.bt.q[1:self.nc,t]

        return ( self.polt.q[1:self.nc,t] - (1-self.polt.s[1:,t]) * self.polt.q[0:self.nc-1, t-1] -
            self.polt.i[1:,t] ) / (0.1 * self.bt.q[1:self.nc,0])




    ###############################################################################################
    ####### Methods used to calibrate and numerically evaluate the elasticity matrix #############

    def solvsysobj(self, vec):
        # solvable system of equations, vec are inputs
        # vecouter are guessed values of derivatives from outer optimization (heuristic)

        ret = np.empty(self.nc,dtype=float)

        # place guesses into matrix (with sym):
        for j in range(0,self.nc-1):
            self.dqdr[j+1,j] = vec[j]
            self.dqdr[j,j+1] = vec[j]
        self.dqdr[self.nc-1,self.nc-3] = vec[self.nc-1]
        self.dqdr[self.nc-3,self.nc-1] = vec[self.nc-1]

        # outside good fraction rising smoothly from zero
        outsidemax = np.sum(self.dqdr[:self.nc,self.nc-1]) / self.dqdr[self.nc-1,self.nc-1]
        for j in range(0,self.nc-1):
            ret[j] = np.sum(self.dqdr[:self.nc,j]) / self.dqdr[j,j] - outsidemax*j/(self.nc-1)

        # last eqn is outside good elast:
        ret[self.nc-1] = np.sum(np.sum( self.r0_deponly[j] * self.dqdr[:self.nc,j]) for j in
            range(self.nc)) / np.sum(self.b.q[:self.nc])  - self.eecalib_targets[1]

        return 10*ret

    def calibeeobj(self, vec, verbose=False):

        # vector vec (on top of ee_calibtargets) completes the demand matrix, place it in the matrix:
        idx = 0
        for j in range(self.nc-3):
            for i in range(j+2,self.nc):
                self.dqdr[i,j] = vec[idx]
                self.dqdr[j,i] = vec[idx]
                idx = idx + 1

        # solve the system
        startval = np.empty(self.nc,dtype=float)
        for j in range(0,self.nc-1):
            startval[j] = -0.5 * self.dqdr[j,j]
        startval[self.nc-1] = -0.25 * self.dqdr[self.nc-3,self.nc-3]

        fsolveoutput = fsolve(self.solvsysobj, startval)
        solnprec = self.solvsysobj(fsolveoutput)
```

```
        # diff provides distance to heuristic objective:
        diff = 0.0
        # non-negativity
        for i in range(self.nc):
            for j in range(self.nc):
                if i != j:
                    diff += 100000 * min(self.ee[i,j],0)

        # declining elasticities with difference in age
        #print(self.dqdr[:self.nc,:self.nc])
        wt = np.empty(self.nc, dtype = float)
        for j in range(self.nc):
            for i in range(self.nc):
                if i==j:
                    wt[i] = 0.0
                    continue
                wt[i] = np.float_power((1-self.eefalloff),np.abs(i-j))
            targsum = np.sum(self.dqdr[:,j]) - self.dqdr[j,j]
            wtsum = np.sum(wt)
            #print('wt',wt, 'targsum',targsum)
            for i in range(self.nc):
                if i==j: continue
                #print('targi',i,targsum * wt[i]/wtsum)
                diff += 10*np.float_power( self.dqdr[i,j] - targsum * wt[i]/wtsum, 2)

        if self.itco % 2000 == 0 or verbose:
            print('Demand system calibration iteration: ',self.itco, ', distance: ', diff)
        self.itco+=1

        return diff

    def ndqdr(self, M0, r0):
        # evaluate forward differences demand Jacobian at M0 and r0
        eps = 1e-10
        J = np.zeros([len(r0), len(r0)], dtype = float)
        q0 = self.dem(M0, r0)
        for i in range(len(r0)):
            r1 = r0.copy()
            r1[i] = r1[i] * (1.0+eps)
            q1 = self.dem(M0, r1)
            J[ : , i] = (q1 - q0) / (r1[i]-r0[i])
        return J

    def checkcalib(self):
        # check (numerically) that calibration has reproduced baseline quantities and elasticities
        numq = self.dem(self.M[0], self.b.r)
        print('-- Check baseline calibration --')
        print('baseline demand input')
        print(self.b.q)
        print('baseline price input')
        print(self.b.p)
        print('baseline r')
        print(self.b.r)
        print('demand evaluated at baseline r')
        print(numq)
        print('theta matrix')
        print(self.tt)
        print('demand derivatives (finite differences) at baseline q and r in output files')
        cats = np.arange(self.ng)
        numee, numdqdr = self.catelast(self.ng,cats,dqdr=True,outfiles=True)

    def catelast(self, n, cats, outfiles=False, dqdr=False):
        # numerically estimate elasticities when changing prices of more than one age at a time
        # cats is a vector containing integers 0 through n-1 that associate each age with a category
        # reported elasticies are with respect to depreciation component of rental price
        eps = 1e-7
        r0n = np.zeros(n, dtype = float)
        r0n_deponly = np.zeros(n, dtype = float)
```

```
        q0n = np.zeros(n, dtype = float)
        q1n = np.zeros(n, dtype = float)
        numdqdr = np.zeros([n, n], dtype = float)
        numee = np.zeros([n, n], dtype = float)

        r0 = self.b.r.copy()
        r0_deponly = r0.copy()
        q0 = self.dem(self.M[0], r0)

        # get depreciation component for producing elasticities independent of repair spending:
        for a in range(self.nc-1):  # (for the oldest car we already have only depreciation since car is
            no longer repaired)
            r0_deponly[a] = self.b.p[a] - (1 - self.b.s[a+1])*self.b.p[a+1]/(1 + self.dd)

        for i in range(n):
            r0n[i] = np.average(r0, weights=((cats==i)*q0))
            r0n_deponly[i] = np.average(r0_deponly, weights=((cats==i)*q0))
            q0n[i] = ((cats==i)*q0).sum()

        # loop through categories
        for i in range(n):
            r1 = r0 + (cats==i) * r0 * eps  # increment prices of all vehicles in category i
            q1 = self.dem(self.M[0], r1)
            r1n = np.average(r1, weights=((cats==i)*q0))
            for j in range(n):
                q1n[j] = ((cats==j)*q1).sum()
            numdqdr[ : , i] = (q1n - q0n) / (r1n-r0n[i])
            numee[ : , i] = numdqdr[ : , i] * r0n_deponly[i] / q0n

        if outfiles==True:
            np.savetxt("eedeponly_cats"+str(n)+".csv", numee, delimiter=",")
            np.savetxt("dqdr_cats"+str(n)+".csv", numdqdr, delimiter=",")
        if dqdr==True:
            return numee, numdqdr
        return numee

    def fleetelasticity(self):
        cats = np.zeros(self.ng, dtype = float)
        cats[-1] = 1
        return self.catelast(2,cats)

    def twoageelasticity(self,dqdr=False):
        cats = np.zeros(self.ng, dtype = float)
        cats[0] = 0
        cats[1:-1] = 1
        cats[-1] = 2
        return self.catelast(3,cats,dqdr=dqdr)

    def numelast(self):
        # calculate and print a variety of numerical elasticities of demand
        cats = np.zeros(self.ng, dtype = float)

        # fleet elasticity
        numee = self.fleetelasticity()
        print('Fleet elasticity')
        print(numee)

        # all ages separately, write file
        cats = np.arange(self.ng)
        numee = self.catelast(self.ng,cats,outfiles=True)

        # newused, write file
        cats = np.zeros(self.ng, dtype = float)
        cats[0] = 0
        cats[1:-1] = 1
        cats[-1] = 2
        numee = self.catelast(3,cats,outfiles=True)
        print('Numerical elasticities newused')
        print(numee)
```

```python
    def set_dqdr(self,i,j,val):
        self.dqdr[i,j] = val
        self.ee[i,j] = self.dqdr[i,j] * self.b.r[j] / self.b.q[i]
        if i==j:
            self.tt[i,j] = (self.ee[i,j] + 1) * self.b.r[i] * self.b.q[i] / self.M[0]
        else:
            self.tt[i,j] = self.ee[i,j] * self.b.r[i] * self.b.q[i] / self.M[0]

    def set_ee(self,i,j,val):
        self.ee[i,j] = val
        self.dqdr[i,j] = self.ee[i,j] / (self.b.r[j] / self.b.q[i])
        if i==j:
            self.tt[i,j] = (self.ee[i,j] + 1) * self.b.r[i] * self.b.q[i] / self.M[0]
        else:
            self.tt[i,j] = self.ee[i,j] * self.b.r[i] * self.b.q[i] / self.M[0]

    def set_tt(self,i,j,val):
        self.tt[i,j] = val
        if i==j:
            self.ee[i,j] = (self.tt[i,j] * self.M[0]) / (self.b.r[i] * self.b.q[i]) - 1
        else:
            self.ee[i,j] = (self.tt[i,j] * self.M[0]) / (self.b.r[i] * self.b.q[i])
        self.dqdr[i,j] = self.ee[i,j] / (self.b.r[j] / self.b.q[i])



######## General functions ########

# Convert quantities by age (q) to and from model year quantities (myq) as in EMFAC
# frac is the fraction of a new vintage sold before the snapshot observation of the vehicle stock occurs
# if frac = 1 then model year quantities and age-based quantities are the same
def myq_to_q(myq, nc, frac=0.89):
    q = np.zeros(nc, dtype=float) # quantities converted to a model year framework
    q[0] = myq[0] + (1-frac)*myq[1]
    for j in range(1,nc-1):
        q[j] = frac*myq[j] + (1-frac)*myq[j+1]
    q[nc-1] = frac*myq[nc-1]
    return q

def q_to_myq(q, nc, frac=0.89):
    myq = np.zeros(nc, dtype=float) # quantities converted to a model year framework
    myq[nc-1] = q[nc-1]/frac
    for j in reversed(range(1,nc-1)):
        myq[j] = (q[j] - (1-frac)*myq[j+1])/frac
    myq[0] = q[0] - (1-frac)*myq[1]
    return myq
```