Fungal lignin peroxidase does not produce the veratryl alcohol cation radical as a diffusible ligninolytic oxidant

Carl J. Houtman¹, Eranda Maligaspe#¹, Christopher G. Hunt¹, Elena Fernández-Fueyo², Angel T. Martínez³, and Kenneth E. Hammel¹*²

From the ¹US Forest Products Laboratory, Madison, WI 53726; ²Centro de Investigaciones Biológicas, Consejo Superior de Investigaciones Científicas, Madrid, Spain; ³Department of Bacteriology, University of Wisconsin, Madison, WI 53706

Running title: Veratryl alcohol radical not released by lignin peroxidase

#Present address: Department of Chemistry, St. John’s University, Queens, NY 11439

¹Present address: Department of Biotechnology, Delft University of Technology, Delft, The Netherlands

*To whom correspondence should be addressed: Kenneth E. Hammel: US Forest Products Laboratory, Madison, WI 53726; kehammel@wisc.edu; Tel. (608) 231-9528. Fax. (608) 231-9262.

Keywords: Lignin degradation, Phanerochaete chrysosporium, biodegradation, peroxidase, free radicals, ligninolytic enzyme, white rot

ABSTRACT

Peroxidases are considered essential agents of lignin degradation by white-rot basidiomycetes. However, low-molecular-weight oxidants likely have a primary role in lignin breakdown because many of these fungi delignify wood before its porosity has sufficiently increased for enzymes to infiltrate. It has been proposed that lignin peroxidases (LiPs, EC 1.11.1.14) fulfill this role by oxidizing the secreted fungal metabolite veratryl alcohol (VA) to its aryl cation radical (VA•+), releasing it to act as a one-electron lignin oxidant within woody plant cell walls. Here, we attached the fluorescent oxidant sensor BODIPY 581/591 throughout beads with a nominal porosity of 6 kDa and assessed whether peroxidase-generated aryl cation radical systems could oxidize the beads. As positive control, we used the 1,2,4,5-tetramethoxybenzene (TMB) cation radical, generated from TMB by horseradish peroxidase. This control oxidized the beads to depths that increased with the amount of oxidant supplied, ultimately resulting in completely oxidized beads. A reaction–diffusion computer model yielded oxidation profiles that were within the 95% confidence intervals for the data. By contrast, bead oxidation caused by VA and the LiPA isozyme of Phanerochaete chrysosporium was confined to a shallow shell of LiP-accessible volume at the bead surface, regardless of how much oxidant was supplied. This finding contrasted with the modeling results, which showed that if the LiP/VA system were to release VA•+, it would oxidize the bead interiors. We conclude that LiPA releases insignificant quantities of VA•+ and that a different mechanism produces small ligninolytic oxidants during white rot.

The oxidative degradation of lignin by white rot basidiomycetes that decay wood is an important component of global carbon cycling, but remains poorly understood in mechanistic detail. On one hand, there is considerable evidence that specialized peroxidases have an important role. In particular, a subset of them termed lignin peroxidases (LiPs, EC 1.11.1.14) cleaves the principal nonphenolic structures of lignin regioselectively (1-3). Moreover, the LiPs, along with structurally related Mn(II)-oxidizing peroxidases (manganese peroxidases, EC 1.11.1.13; and versatile peroxidases, EC 1.11.1.16) are unique to the genomes of most white rot fungi
Veratryl alcohol radical not released by lignin peroxidase

while being absent from other fungal genomes (1, 4, 5).

On the other hand, many white rot fungi begin to delignify the wood cell wall before its porosity has increased enough for enzymes to infiltrate, in the process removing lignin faster than they remove polysaccharides (1, 6, 7). This observation of selective delignification suggests that for many fungi the proximal oxidants of lignin, at least during early white rot, are low molecular weight agents rather than enzymes. However, the only small oxidants that white rot fungi indisputably produce are Mn(III) chelates, which are generated by some of the abovementioned peroxidases but are insufficiently reactive to attack nonphenolic lignin structures (2, 3, 8-10).

It has been proposed that some white rot fungi circumvent this problem by using LiPs to oxidize secreted low molecular weight metabolites to reactive free radicals, which might then diffuse into the wood cell wall and cleave lignin oxidatively (11, 12). LiPs are remarkable among peroxidases in their efficient oxidation of nonphenolic aromatic compounds that resemble the major nonphenolic structures of lignin (2), although the versatile peroxidases also have this ability to some extent (8,9). Substrate oxidation occurs by a classic heme peroxidase mechanism in which compounds I and II of the enzyme catalyze single electron transfers (SETs, Fig. 1) (13), which are mediated by a solvent-exposed tryptophan residue (14-16). When LiPs react with nonphenolic lignin structures, the immediate oxidation products are aryl cation radicals that spontaneously fragment between C1 and C2 of the lignin propyl sidechain. This oxidative sequence is thought to be the route by which LiPs catalyze ligninolysis directly when diffusible mediators are not involved (17, 18).

The chief candidate for a mediator that would operate in conjunction with LiP is veratryl alcohol (VA, 3,4-dimethoxybenzyl alcohol), a secondary metabolite secreted by many white rot fungi (19). The best-known species is Phanerochaete chrysosporium, some isolates of which probably employ low molecular weight oxidants because they have been observed to remove lignin from wood faster than they remove cellulose (6). LiP oxidizes VA to veratraldehyde (3,4-dimethoxybenzaldehyde) by the same SET mechanism that it employs on nonphenolic lignin structures (13), producing the veratryl alcohol cation radical (VA⁺) as an intermediate that has been detected by ESR spectroscopy (20). Conceivably, VA⁺ might escape into solution and then extend the spatial range of LiP activity by acting as a SET oxidant of occluded lignin structures (11, 12).

In support of this hypothesis, VA⁺ is stable enough in water to diffuse several micrometers, hydrolyzing with a rate constant of 17 s⁻¹ (11, 21, 22). Furthermore, the oxidation of VA by LiP in air at physiological pH (4.5) consumes O₂ and produces approximately 1.1 veratraldehyde per H₂O₂ supplied (23). This finding suggests that some VA⁺ may escape oxidation at the LiP active site, hydrolyzing instead to give benzylic radicals that rapidly add O₂. The resulting α-hydroxyperoxyl radicals would in turn eliminate the H₂O₂ precursor HO₂•, thus accounting for the enhancement in veratraldehyde yield (24). Finally, VA has been shown to enhance some LiP-catalyzed oxidations (12, 25-27), and is required for LiP to depolymerize colloidal synthetic lignins (28). In some cases, there is good evidence that this stimulation results from mediation by VA⁺. For example, VA is required for LiP to oxidize 4-methoxymandelic acid, which quenches the ESR signal of the VA⁺ produced, and veratraldehyde production during these reactions occurs only after all of the 4-methoxymandelic acid has been consumed. Moreover, the rate of 4-methoxymandelic acid oxidation by LiP exhibits saturation kinetics as the concentration of VA is increased (27).

On the other hand, none of the above observations establish that any of the VA⁺ produced during LiP turnover escapes into free solution. An alternative explanation is that SET from a substrate donor to LiP could be mediated by VA⁺ that remains on the enzyme surface (27, 29). In this case, the physiological and perhaps critically important function of the bound VA⁺ would be to facilitate contact and electron transfer between the bulky, stereoirregular structures of lignin and oxidized LiP, as opposed to acting as a diffusible, remote oxidant of lignin. This concept
Veratryl alcohol radical not released by lignin peroxidase

of VA•• as a bound mediator was introduced because one research group reported that the rate constant for hydrolysis of VA•• is too high for it to diffuse a significant distance (29), and also because chemically generated VA•• failed to oxidize 4-methoxymandelic acid (12). To reconcile these observations with the enzyme kinetics experiments that support mediation, it was hypothesized that a LiP/VA•• complex forms during catalysis and is a stronger and more long-lived oxidant that enables the oxidation of recalcitrant substrates (27, 29).

However, VA•• must be able to act as a diffusible oxidant if it is released, because chemically generated VA•• was shown to oxidize a polymeric dye (12). Moreover, multiple studies confirmed that the hydrolysis rate constant for VA•• is in fact relatively low at 17 s⁻¹ (11, 21, 22). The problem appears to be that the original research conflated two issues: how strong an oxidant VA•• is and whether LiP releases it into solution. Certainly the redox potential of VA••, a dimethoxylated aromatic, is relevant to the oxidation of the monomethoxylated aromatic 4-methoxymandelic acid, whose reaction with VA•• is expected to be thermodynamically unfavorable. However, this constraint does not apply to the oxidation of wood lignin, whose rings carry two or three methoxyl groups, and thus the experiments with 4-methoxymandelic acid and other recalcitrant substrates do not answer the remaining question of whether VA•• can escape the LiP active site.

The persistence of this question in the literature, and its importance for our understanding of which small diffusible oxidants white rot fungi actually employ, led us to conclude that an empirical approach was needed: if some of the VA•• produced by LiP can escape the enzyme and oxidize a donor at a distance, then it should be possible to observe the oxidation of a non-recalcitrant substrate immobilized within a porous material that excludes LiP (42 kDa) while admitting VA•• (168 Da). As the donor substrate, we used the ratiometric oxidation sensor BODIPY 581/591, which undergoes facile oxidation, and whose fluorescence shifts from red to green once its conjugated diene sidechain has been oxidatively cleaved (30). BODIPY 581/591 is directly oxidized by LiP, and VA markedly stimulates the reaction by a mechanism not yet determined (31). To make most of the dye inaccessible to direct enzymatic oxidation, we attached it covalently throughout porous polyacrylamide beads that largely exclude LiP. In the presence and absence of VA, we treated the beads with LiPA, one of the most abundant LiP isozymes of P. chrysosporium and the only one that has been investigated in mechanistic detail (3, 32). We then used confocal fluorescence microscopy to determine the spatial distributions of BODIPY oxidation in the beads, comparing the results with computer simulations of the profiles expected if free VA•• is the oxidant. In effect, our fluorescent beads serve as a rough model of the similarly semi-porous, lignocellulosic wood cell wall.

Results

Properties of dye-functionalized beads

The upper panels of Figure 2 show red channel confocal fluorescence images of a 92-µm diameter polyacrylamide bead containing pores with a nominal 6 kDa molecular mass exclusion, after covalent attachment of BODIPY 581/591 via amide linkages. The distribution of dye throughout the bead is fairly uniform, although a decrease at its center can be discerned. This inhomogeneity in the beads was acceptable for our purposes because the method for measuring dye oxidation is ratiometric, and thus independent of dye concentration.

Close inspection of the bead’s edge additionally shows that some BODIPY-labeled polymer extends 1-2 µm into the surrounding solution. The lack of a sharp edge is expected because adsorption of the surfactant used when the beads are manufactured by inverse phase suspension polymerization leads to the formation of larger pores near their surfaces (33, 34). However, this feature leads to an uncertainty of less than 0.25 µm in the position of the edge (see Experimental Procedures).

Before using the beads in oxidation experiments, we made a preliminary estimate of how deeply LiPA (42 kDa) could penetrate. This assessment was necessary because beads produced by inverse phase polymerization do not exhibit an
Veratryl alcohol radical not released by lignin peroxidase

absolute molecular mass exclusion limit, but rather contain a gradient of porosity that decreases from the edge to the core (33, 34). Consequently, the beads were expected to contain a peripheral zone where direct oxidation of the BODIPY by LiP could occur without the participation of a diffusible mediator.

To estimate the size of this zone, we suspended beads in a solution of fluorescein-labeled ovalbumin (45 kDa) and obtained both red channel images to visualize the beads (Fig. 2, upper) and green channel images to visualize the surrounding solution (Fig. 2, lower). A comparison of the images reveals a gradient of green pixels due to fluorescein that diminishes inwards from the edge of the bead. Adjacent to the edge, this green emission appears more intense than it actually is, because the volume sampled by the microscope is not small enough to avoid capturing some of the extremely bright fluorescence from the surrounding solution. However, within the bead the green pixels indicate the presence of the ovalbumin probe.

To quantify this gradient, we conducted image analyses on nine replicate beads. The data are represented as green fluorescence intensity relative to the intensity in the solution surrounding the beads (Fig. 3), and show that 95% of the 45 kDa protein was excluded at a depth of 7 µm. This result suggested that the beads could serve our purpose, provided infiltration of their outermost volume by LiP was taken into account.

Proof of principle for mediated bead oxidation

As a test of whether BODIPY 581/591 molecules immobilized within the derivatized beads could detect small, diffusible cation radical oxidants, we began with the oxidation of 1,2,4,5-tetramethoxybenzene (TMB), a synthetic compound that white rot fungi do not secrete, by LiP and by the similarly sized enzyme horseradish peroxidase (HrP, 44 kDa). Both enzymes oxidize TMB to its cation radical (TMB +•), which is released into solution and hydrolyzes slowly with a half-life of minutes (35, 36). TMB +• is not only relatively stable, but is also predicted to be a good oxidant with $E^0 = +1.30$ V in H$_2$O (37). These properties suggest that it could mediate SET oxidations at a distance from the peroxidase that generates it.

In preliminary experiments with LiP, we exposed the dye-functionalized beads to various levels of H$_2$O$_2$ in the presence of excess TMB, waited until spectrophotometric observation at 450 nm showed that TMB +• production had ceased (35, 36), and then examined the beads by confocal fluorescence microscopy. As shown by the representative images of beads in Figure 4A, the initial red emission was replaced from the periphery inwards by green emission, and the beads became entirely green when sufficient H$_2$O$_2$ had been added. Our supply of LiP being limited, we then performed a larger number of quantitative experiments using HrP, exposing the beads to various levels of TMB in the presence of excess H$_2$O$_2$. This experiment revealed the presence of a distinct advancing front of oxidation that progressed towards the bead center as a function of the amount of TMB +• supplied (Fig. 4B), which established that our beads were a suitable reporter for small aryl cation radical oxidants.

Development of a reaction-diffusion model

We next developed a mathematical model for bead oxidation and checked its validity against the experimental results obtained with HrP and TMB. The model handles the varying permeability of the beads to the peroxidase, rate constants for all of the relevant reactions, and diffusion coefficients for all of the moving species. The theory of diffusion and reaction in spherical particles is well developed and provides the basis for our model (38). As there are 11 species to follow (Fig. 1, Table 1) we used a finite-volume numerical method to solve the balance equations simultaneously (Data Set S1).

The model reveals that diffusion rates have a relatively low impact in our system. The diffusion of HrP can be disregarded because it is preloaded into the beads. Although diffusion is a factor for H$_2$O$_2$ and TMB, since they continue to enter the beads as the peroxidase consumes them, this phenomenon has little influence on the shapes of the oxidation profiles because diffusion is rapid relative to the duration of the reaction. This is evident from the characteristic times ($t$) for diffusion of H$_2$O$_2$ and TMB, where $t = R^2/D$, $R$ being the bead’s radius and $D$ being the diffusion coefficient for the substrate. The average value of $R$ for the beads is 35 µm, whereas $D$ is...
Veratryl alcohol radical not released by lignin peroxidase

approximately $5 \times 10^{-6}$ cm$^2$ s$^{-1}$ for H$_2$O$_2$ and $3 \times 10^{-6}$ cm$^2$ s$^{-1}$ for TMB (42, 43). Considering just the case of TMB, the slower-moving substrate, the resulting $t$ of 4 seconds indicates that within 16 seconds of its addition TMB arrives to 98% of the equilibrium concentration it would have in the absence of reactions that consume it (38). Likewise, the TMB$^••$ produced by HrP quickly attains its concentration profile within the bead, as its expected diffusion coefficient is essentially the same as that for TMB. In contrast to this rapid equilibration of all the small molecules in our system, the duration of the reaction for bead oxidation is relatively long at 6 min.

Given the relatively low impacts of diffusion rates, the oxidation profiles in the beads are governed chiefly by the rates for production and reaction of TMB$^••$. TMB$^••$ production by the peroxidase is rapid under our conditions, with a rate constant of $3.96 \times 10^5$ M$^{-1}$ s$^{-1}$ (Fig. S1), and obviously affects the rate of bead oxidation, but it does not affect the shapes of the profiles that appear in partially oxidized beads. Instead, the computer model reveals that the shape of the oxidation profiles is largely determined by the rates for TMB$^••$ reaction with the dye and for TMB$^••$ hydrolysis. For a given rate of reaction with the dye, a higher hydrolysis rate will result in oxidation that penetrates the bead less deeply. For a given hydrolysis rate, a higher rate of reaction with the dye will steepen the advancing front.

The relative rates at which the cation radical oxidizes the dye or hydrolyzes can be assessed by determining the apparent first order rate constants for the two processes. Dye oxidation probably begins with a reversible outer sphere SET to TMB$^••$, which is expected to be a very rapid (diffusion-limited) process with a second order rate constant around $10^9$ M$^{-1}$ s$^{-1}$ (44). However, this initial SET cannot be the rate-limiting step—if it were, plots of fraction green versus radius of the beads would show abrupt oxidation fronts rather than the gradients we observed. It appears that some subsequent irreversible reaction associated with cleavage of the BODIPY 581/591 conjugated diene sidechain is likely rate-limiting (30). To estimate the actual rate constant ($k_9$) for dye oxidation, we iteratively varied it until the model matched the observed oxidation gradients. This approach gave a value of $6 \times 10^5$ M$^{-1}$ s$^{-1}$ for $k_9$ that fits all the data (Table 1). Since the concentration of the dye in the beads was 0.5 mM, the apparent first-order rate constant for reaction of TMB$^••$ with the dye is then 300 s$^{-1}$. By contrast, the rate of TMB$^••$ hydrolysis as determined from our spectrophotometric observations (Fig. S1) is much lower at 0.03 s$^{-1}$.

These two rate constants determine that the advance of green fluorescence into the beads appears as a gradually sloped front: as long as dye molecules are available, most TMB$^••$ molecules react preferentially with them instead of hydrolyzing, but as more of the dye is oxidized an increasing proportion of the TMB$^••$ molecules is able to advance deeper into the bead. The results of the model, shown in Figure 4B, generate predicted oxidation profiles that agree well with the experimental data obtained using HrP and TMB. To assess the statistical significance of this finding, we compared the positions in the experimental and modeled oxidation fronts at which the BODIPY dye was 50% oxidized. The results show that the model is within the 95% confidence intervals for the data (Fig. 4C).

Test of veratryl alcohol as a mediator

Since VA$^••$, with $E^0 = +1.36$ V in H$_2$O (37), is a stronger oxidant than TMB$^••$, the foregoing results suggested that BODIPY 581/591 immobilized within small pores should also serve as a probe to detect VA$^••$ if this cation radical escapes from the LiP that produces it. The principal difference expected with the LiP/VA system is that it would produce a less diffusible oxidation system that exhibits a sharper advancing front, because the rate constant for hydrolysis of VA$^••$ is 17 s$^{-1}$ (11, 21, 22), in contrast to our observed value of 0.03 s$^{-1}$ for TMB$^••$ hydrolysis (Table 1, Fig. S1). To look for evidence of VA-derived diffusible oxidants, we began by comparing the spatial distribution of oxidized dye in beads that had been oxidized by LiP with and without VA. Figure 5A shows representative images of the beads, and Figure 5B shows the fraction of green intensity versus distance from the surface for replicate beads. The data show that, although VA stimulated the extent of BODIPY dye oxidation at the bead periphery as expected (31), the depth of the reaction was no different
with and without VA. In both cases, the extent of oxidation 7 µm into the bead was about 5% of the extent observed at the periphery. The close correspondence between this result and the depth of labeled ovalbumin infiltration into the beads (Fig. 3) suggests that dye oxidation occurred only in the space available to LiP.

However, it was necessary to consider an alternative possibility, namely that the depths of bead oxidation with and without VA were the same simply because the amount of H$_2$O$_2$ added was insufficient under our reaction conditions. Therefore, we performed additional experiments with LiP in the presence of excess VA, using a series of increasing H$_2$O$_2$ concentrations. A technical difficulty with this approach is that some LiP inactivation is unavoidable at high levels of H$_2$O$_2$ (45). Nevertheless, we were able to obtain progressively increasing extents of veratraldehyde production in the range of 0.5-4.0 mM H$_2$O$_2$ (Table S1). If a significant fraction of the VA$^{+•}$ molecules that are precursors for this veratraldehyde can escape from the enzyme, it follows that the depth of bead oxidation should increase as the veratraldehyde yield increases.

The image analysis of beads treated according to this regime (Fig. 6A and 6B) shows that oxidation initially extended deeper into the LiP-accessible periphery of the beads as the H$_2$O$_2$ concentration was raised from 0.5 mM to 1.0 mM. However, no deeper advance of the oxidation front into the LiP-inaccessible region was obtained when the H$_2$O$_2$ concentration was increased progressively to 4.0 mM, even though the veratraldehyde yields indicate that the additional H$_2$O$_2$ gave a 3.6-fold increase in the number of LiP turnovers. Moreover, the final depth of oxidation appeared no greater than that obtained via direct oxidation by LiP without VA in the Figure 5 experiment. The only apparent change in the beads treated with the highest level of H$_2$O$_2$ was that the intensity of their peripheral green fluorescence dropped slightly, perhaps due to bleaching of the oxidized BODIPY dye via additional LiP-catalyzed reactions.

This result contrasts with the data obtained using TMB as the substrate, where TMB$^{+•}$ clearly acts as a diffusible mediator (Fig. 4), but the question remains as to how the oxidation profiles would appear if the VA$^{+•}$ produced by LiP did escape and diffuse into the beads. Specifically, is the relatively high rate of VA$^{+•}$ hydrolysis by itself sufficient to explain why supplying VA did not increase the depth of bead oxidation by LiP? To address this question, we applied our computer model, as it had given a good fit between data and theory for bead oxidation mediated by TMB$^{+•}$. The apparent rate constant for oxidation of the BODIPY dye by VA$^{+•}$ ($k_9$, Table 1) was taken to be the same as that for oxidation of the dye by TMB$^{+•}$, since it appears that the initial SET oxidation of the dye is in any case not rate-limiting. Several of the other rate constants were changed to match independent experimental data that apply to VA$^{+•}$ and LiP (Table 1, Fig. S2), but the only one of these to have a large effect of the performance of the model was the VA$^{+•}$ hydrolysis rate constant ($k_7$), which was set per the literature at 17 s$^{-1}$ (11, 21, 22).

The model assumes that 10% of LiP turnovers release VA$^{+•}$ into solution, thus accounting for published results on the stoichiometry of veratraldehyde production by the enzyme (23). The results are included with the experimental data in Figure 6B, and it is clear that there is a marked difference between the observed and modeled depths of oxidation. At 0.5 mM H$_2$O$_2$, the fit between the model and data is very good, which validates the value of $k_9$ we carried over from the HrP/TMB model. However, at higher H$_2$O$_2$ concentrations, the model indicates a continued advance of oxidation into the bead, in contrast to the lack of deeper oxidation found experimentally. For statistical analysis, we again compared the observed and modeled positions in the advancing oxidation fronts at which the BODIPY dye was 50% oxidized (Fig. 6C). Although the depth of 50% oxidation increased in the LiP-accessible periphery when the H$_2$O$_2$ concentration was raised from 0.5 mM to 1.0 mM, the data do not support deeper oxidation into the LiP-inaccessible volume with increasing levels of H$_2$O$_2$. Instead, the depth of 50% oxidation appeared to approach an asymptotic value near 4.3 µm.
Veratryl alcohol radical not released by lignin peroxidase

Discussion

VA did not enable LiPA, the major isozyme of the LiP H8 family (3, 32), to oxidize BODIPY 581/591 that was immobilized within beads containing pores too small for the enzyme to infiltrate. Given the marked disparity between the data and the computer simulations, the observed oxidation profiles within the beads cannot be attributed to the diffusion and reaction of VA+•. Instead, the profiles fit the progressively lower porosity that occurs with increasing depth into the beads, such that dye oxidation was possible only where LiP was present. We conclude from these results that LiPA does not deploy VA+• to act as a diffusible oxidant. Of course, a role for VA+• as a SET mediator that remains bound to LiP continues to be a valid hypothesis, and appears consistent with our observation that VA enhanced dye oxidation by LiP at the bead periphery.

A pertinent question is whether any of the other nine LiPs encoded in the *P. chrysosporium* genome might employ VA+• as a remote oxidant. A definitive answer is not yet available, because none of these isozymes were available for this study, and none have been investigated in enough mechanistic detail to obtain the kinetic constants needed for our reaction-diffusion model. However, it is worth noting that all of the LiP isozymes have similar steady-state catalytic properties towards VA (46), and all display a high degree of sequence identity (Fig. S3) (3). Moreover, although computational simulations of VA+• binding to LiP have not yet been reported, comparisons of the VA-binding surface environment, which comprises 20 amino acid residues near the catalytic tyrosine residue (W171 in LiPA), indicate only small variations (Fig. S4). (14, 47). In particular, most of the acidic and aromatic residues that might interact with an aryl cation radical are conserved. Thus, although VA+• in free solution is a reasonable oxidant of lignin from a chemical standpoint, the available data indicate that LiPs are unlikely to produce it. Additional work with other LiP isozymes is certainly needed, but the explanation for initial attack by *P. chrysosporium* on lignin in the cell wall of sound wood probably lies elsewhere.

The best available alternative hypothesis involves sequential action by two secreted enzymes whose genes are transcribed at high levels during the growth of *P. chrysosporium* on wood: cellobiose dehydrogenase and manganese peroxidase (32). In the first step, the former enzyme reduces Fe(III), thus generating Fe(II) chelates that drive Fenton chemistry (48, 49, 50). Autoxidation of some of this Fe(II), as well as the action of fungal oxidases on secreted metabolites such as glyoxal, generate the H2O2 that is also required (1, 32). Since Fenton oxidation of lignin is expected to result in the O-demethylation of some of its aromatic rings, the polymer is predicted to become enriched in phenolic substructures (51, 52). These phenolic units, unlike the nonphenolic aryl ether units that predominate in native lignin, can be oxidatively cleaved by Mn(III) chelates, which are generated by manganese peroxidase in the second step (2, 3, 50). Although this sequence of events remains to be demonstrated for wood undergoing white rot, there is no question that the proposed reagents–Fe(II) chelates, H2O2, and Mn(III) chelates–are all small enough to infiltrate the wood cell wall (53).

Experimental Procedures

**Synthesis of dye-functionalized beads**

Porous polyacrylamide beads with a nominal 6 kDa exclusion limit (Bio-Gel P-6 fine, Bio-Rad, Hercules, CA) were used. They had a mean diameter of 72 ± 14 µm and a 48-104 µm diameter range (n = 150). Preparatory to BODIPY attachment, 2 g dry wt of the beads was first aminoethylated by the method of Inman and Dintzis (54). The derivatized beads were found by conductimetric titration to contain 245 µmol of NH2 groups per g dry wt, and were stored in water/ethanol, 4:1 at 4°C.

Aminoethylated beads (100 µl settled vol, 1.3 µmol NH2 groups assuming random close packing, defined as 1 equivalent) were transferred to an Eppendorf centrifuge filter (0.4 µm pore size) and equilibrated with 0.1 M NaHCO3 (pH 8.2, 100 µl) by several cycles of low speed centrifugation (700 rpm, 1 min), with repeated addition of the NaHCO3 solution. After a final centrifugation step, 0.17 equivalent of BODIPY 581/591 C11 SE (4,4-difluoro-5-(4-phenyl-1,3-butadienyl)-4-bora-3a,4a-diaza-s-indacene-3-undecanoic acid succinimidyl ester, Thermo Fisher...
Veratryl alcohol radical not released by lignin peroxidase

Scientific, Waltham, MA) was added in sufficient N,N-dimethylformamide to wet the beads throughout. The mixture was allowed to react for 3 h at room temperature and then stored at 4°C overnight. Removal of the supernatant fraction by centrifuge filtration showed that it was colorless, whereas the beads were bright red, indicating that all the dye had attached to them. The final inferred concentration of amide-linked BODIPY 581/591 in the beads was 0.5 mM.

The above derivatization procedure was then repeated using 2 equivalents of the succinimidyl ester of carboxy-terminated 40 kDa polyethylene glycol monomethyl ether (Nanocs, New York, NY). This step was included because we found that it somewhat decreased the permeability of the beads’ outer surface to proteins. Finally, any remaining NH₂ groups on the beads were capped by a third cycle of derivatization with 10 equivalents of succinimidyl acetate (TCI America, Portland, OR), using the above procedure. The beads were rinsed with water and stored in water/ethanol, 4:1 at 4°C.

**Enzymes and reagents**

Horseradish peroxidase (HrP) was Type VI from Sigma-Aldrich (St. Louis, MO). 1,2,4,5-Tetramethoxybenzene was a generous contribution from P. J. Kersten (Forest Products Laboratory, Madison, WI). Veratryl alcohol (VA, 3,4-dimethoxybenzyl alcohol, Sigma-Aldrich) was purified by vacuum distillation as described earlier (55).

LiPA was expressed heterologously in *Escherichia coli* W3110. The mature protein-coding sequence of the encoding gene (GenBank Y00262; corresponding to LiPA in the *P. chrysosporium* genome, with JGI ID# 2989894) was manually curated and then synthesized by ATG:biosynthetics (Merzhausen, Germany), after checking whether all the codons had previously been used to express other genes in *E. coli* W3110, and substituting them where required. This sequence was cloned in the expression vector pFLAG1 (International Biotechnologies, Cambridge, UK) and the resulting plasmid pFLAG1-Y00262 was directly used for expression. *E. coli* DH5α was used for plasmid propagation.

To produce the LiP, cells of transformed *E. coli* W3110 were grown for 3 h in Terrific Broth (Sigma-Aldrich), induced with 1 mM isopropyl-β-D-thiogalactopyranoside, and grown further for 4 h. The apoenzyme accumulated in inclusion bodies, as observed by sodium dodecyl sulfate-polyacrylamide gel electrophoresis (SDS-PAGE), and was solubilized using 8 M urea. Subsequent *in vitro* refolding of the LiP was performed using 2.1 M urea, 5 mM Ca²⁺, 10 µM hemin, 0.7 mM oxidized glutathione, 0.1 mM dithiothreitol, and 0.2 mg ml⁻¹ protein, at pH 8.0 (56). Active enzyme was purified by Resource-Q chromatography (GE Healthcare, Chicago, IL), using a 0-300 mM NaCl gradient (20 min, 2 ml min⁻¹ flow rate) in 10 mM sodium tartrate (pH 5.5) containing 1 mM CaCl₂.

Previous work has shown that the kinetic properties of heterologously expressed LiPA from *P. chrysosporium* are virtually the same as those of LiPA isolated from the fungus (1, 41).

**Enzymatic oxidations of dye-functionalized beads**

Oxidations were carried out in 1-ml Eppendorf tubes at room temperature, in a final vol of 200 µl. When the role of 1,2,4,5-tetramethoxybenzene (TMB) as a mediator for HrP was being investigated, complete reactions contained HrP (0.3 µM), H₂O₂ (286 µM), and fluorometric beads (16 µl settled vol) in Na tartrate buffer (20 mM, pH 4.5). After a 1-h preincubation, the reaction was initiated with 0.71 µM TMB and the tube was mixed by continuous vortexing. At 6-min intervals, a 4-µl aliquot containing beads was withdrawn for analysis and an additional 0.71 µM of TMB was added in 4 µl of water. Spectrophotometric monitoring at 450 nm of a parallel reaction without beads indicated that 6 min was sufficient to reach an endpoint after each addition of TMB, *i.e.*, that production of the TMB cation radical (TMB⁺) had ceased before this time (Fig. S1) (35, 36). We used limiting TMB in these experiments because the low quantity of oxidant necessary for bead oxidation dictated that the limiting substrate had to be added at a low concentration, and this would have been difficult with the needed accuracy using H₂O₂, since dilute solutions of it are susceptible to reaction with trace metals.

When the role of TMB or VA as a mediator for LiP was being investigated, complete reactions
(200 µl in Eppendorf tubes) contained LiP (3 µM), TMB (4 mM) or VA (12 mM), and fluorometric beads (3 µl settled vol) in Na tartrate buffer (20 mM, pH 4.5). After a 1-h preincubation, the reactions were initiated with H₂O₂ (0.1-4.0 mM) and the tubes were continuously mixed by vortexing for 30 min. Spectrophotometric monitoring at 310 nm of parallel reactions without beads indicated that the reactions had reached an endpoint before this time, i.e., that veratraldehyde production had ceased (Fig. S2) (51). We used limiting H₂O₂ in these experiments because the relatively high quantity of oxidant necessary for bead oxidation dictated that the nonlimiting substrate had to be added at a concentration much higher than 1 mM. Since LiP is inactivated by millimolar concentrations of H₂O₂, and donor substrates such as VA partially protect the enzyme against this inactivation (45), a design using excess electron donor appeared most prudent.

**Image analysis**

Images were processed using Python 2.7 scripts, based on functions in the OpenCV module, version 2.4.10 (www.opencv.org). The script is available in Data Set S1. For experiments on BODIPY dye oxidation, the edge of the bead was found by selecting a threshold value near 20% of the maximum fluorescence intensity, and then applying it to a gray-scale composite image. This approach leads to an uncertainty of less than 0.25 µm in the position of the edge. The beads were then isolated using a contouring routine. Beads that touched the edge of the image or another bead were excluded from the analysis. From the contour, the centroid of the bead was determined. The pixels 2 to 5 µm outside the bead were used to determine the background emission for each color, and these background values were subtracted from the respective channels. Next, since a range of laser powers had been used to acquire the images, the apparent intensities of all images were adjusted to bring all the data to constant laser power. Finally, the average fraction of green intensity was tallied for all the pixels in a series of annular disks centered on the centroid of the beads. These data are presented as fraction of green versus distance from the center of the beads.

In the case of the fluorescein-labeled ovalbumin experiments, we defined the bead edge as the location at which the red channel fluorescence intensity due to native BODIPY dye was 50% of the background intensity in the surrounding solution. This approach leads to an uncertainty of less than 0.25 µm in the position of the edge.

**Computer modeling**

Reaction-diffusion modeling of the beads was accomplished using Python 2.7 and the FiPy 3.1 module (https://www.ctcms.nist.gov/fipy/), which is an implementation of the finite volume method for solving time-dependent partial differential equations.
differential equations (Data Set S1). Since the beads are spherical, the mathematical model reduces to a time-dependent, one-dimensional set of equations. The chemical reactions allowed are shown in Figure 1, and relevant rate constants are shown in Table 1. The reaction-diffusion model followed 11 species during the course of the reactions: \( E_0 \), ferric resting state peroxidase; \( O=E_i \), two-electron-oxidized peroxidase compound I; \( HO-E_{ii} \), one-electron-oxidized peroxidase compound II; \( HO-E_{ii}-M^+• \), peroxidase compound II associated with mediator cation radical; \( M = \) mediator (TMB or VA); \( M^+• = \) mediator cation radical; \( A \), two-electron-oxidized product derived from mediator (2,5-dimethoxybenzoquinone or veratraldehyde); \( R \), native red BODIPY dye; \( G \), oxidized green BODIPY dye; \( H_2O_2 \); and \( HO_2• \), perhydroxyl radical. As indicated in Figure 1, \( HO_2• \) production, although expected at some level following VA oxidation (23, 24), is not associated with TMB oxidation. Therefore, \( HO_2• \) production was preset to zero when TMB oxidation was modeled.

Because the BODIPY dye was covalently attached to the beads and the peroxidases were pre-equilibrated with them before the reactions, only \( M \), \( M^+• \), \( H_2O_2 \), and \( HO_2• \) were allowed to diffuse in the model. The diffusion coefficient \( (D) \) used for \( H_2O_2 \) was the measured literature value \( (5 \times 10^{-6} \text{ cm}^2 \text{ s}^{-1}) \) (43), and \( D \) for the similarly sized \( HO_2• \) molecule was assumed to be the same. The value of \( D \) for \( M \) and \( M^+• \) was assumed to be the same as reported earlier for similarly sized aromatic molecules in water \( (3 \times 10^{-6} \text{ cm}^2 \text{ s}^{-1}) \) (42). For purposes of the model, the boundary condition at the surface was set to values obtained from a solution phase model assuming that the liquid phase was well mixed. The solution phase model resulted in a set of ordinary differential equations that were solved using the “Integrate” functions in the SciPy module (Data Set S1). Consistent with the spherical symmetry, the boundary condition at the center was the gradient with distance set to zero.

Although the BODIPY dye was not allowed to diffuse in the model, its concentration was allowed to vary across the bead, being set to match measured values that were determined by image analysis of the control beads. In this way, the model took account of the fact that the concentration of attached dye was slightly higher at the bead periphery than at its center. Likewise, although the peroxidase was not allowed to diffuse in the model, its concentration across the bead was allowed to vary, being set to match the profiles of bead oxidation that we observed when LiP was used to oxidize the beads directly without a mediator. That is, the shape of the gradient in dye oxidation was taken to indicate the shape of the gradient in peroxidase infiltration, this approach being feasible because the dye was only partially oxidized, even at the outmost edge of the bead, in experiments without mediator. We used these measurements of dye oxidation as a reporter for peroxidase location because the results were essentially unaffected by background fluorescence, as opposed to the data from measurements of infiltration using fluorescently labeled ovalbumin, which contained signal from the high concentration of labeled ovalbumin that was present in the buffer surrounding the beads.
Conflict of interest
The authors declare that they have no conflicts of interest with the contents of this article.

References
1. Fernández-Fueyo, E., Ruiz-Dueñas, F. J., Ferreira, P., Floudas, D., Hibbett, D. S., Canessa, P., Larondo, L. F., James, T. Y., Seelenfreund, D., Lobos, S., Polanco, R., Tello, M., Honda, Y., Watanabe, T., Watanabe, T., San, R. J., Kubicek, C. P., Schmoll, M., Gaskell, J., Hammel, K. E., St John, F. J., Vanden Wymelenberg, A., Sabat, G., BonDurant, S. S., Syed, K., Yadav, J. S., Doddapaneni, H., Subramaninan, V., Lavin, J. L., Oguiza, J. A., Perez, G., Pisabarro, A. G., Ramirez, L., Santoyo, F., Master, E., Coutinho, P. M., Henriссat, B., Lombard, V., Magnuson, J. K., Kües, U., Hori, C., Igarashi, K., Samejima, M., Held, B. W., Barry, K. W., LaButti, K. M., Lapidus, A., Lindquist, E. A., Lucas, S. M., Riley, R., Salamov, A. A., Hoffmeister, D., Schwenk, D., Hadar, Y., Yarden, O., de Vries, R. P., Wiebenga, A., Stenlid, J., Eastwood, D., Grigoriev, I. V., Berka, R. M., Blanchette, R. A., Kersten, P., Martinez, A. T., Vicuña, R., and Cullen, D. (2012) Comparative genomics of Ceriporiopsis subvermispora and Phanerochaete chrysosporium provide insight into selective ligninolysis. Proc. Natl. Acad. Sci. USA 109, 5458-5463
2. Hammel, K. E., and Cullen, D. (2008) Role of fungal peroxidases in biological ligninolysis. Curr. Opin. Plant. Biol. 11, 349-355
3. Martínez, A. T. (2002) Molecular biology and structure-function of lignin-degrading heme peroxidases. Enzyme Microb. Technol. 30, 425-444
4. Floudas, D., Binder, M., Riley, R., Barry, K., Blanchette, R. A., Henriссat, B., Martinez, A. T., Otiliar, R., Spatafora, J. W., Yadav, J. S., Aerts, A., Benoit, I., Boyd, A., Carlsson, A., Copeland, A., Coutinho, P. M., De Vries, R. P., Ferreira, P., Findley, K., Foster, B., Gaskell, J., Glotzer, D., Górecki, P., Heitman, J., Hesse, C., Hori, C., Igarashi, K., Jurgens, J. A., Kallen, N., Kersten, P., Kohler, A., Kües, U., Kumar, T. K. A., Kuo, A., LaButti, K., Larondo, L. F., Lindquist, E., Ling, A., Lombard, V., Lucas, S., Lundell, T., Martin, R., McLaughlin, D. J., Morgenstern, I., Morin, E., Murat, C., Nagy, L. G., Nolan, M., Ohm, R. A., Patyshakulyeva, A., Rokas, A., Ruiz-Dueñas, F. J., Sabat, G., Salamov, A., Samejima, M., Schmutz, J., Slot, J. C., John, F. S., Stenlid, J., Sun, H., Sun, S., Syed, K., Tsang, A., Wiebenga, A., Young, D., Pisabarro, A., Eastwood, D. C., Martin, F., Cullen, D., Grigoriev, I. V., and Hibbett, D. S. (2012) The paleozoic origin of enzymatic lignin decomposition reconstructed from 31 fungal genomes. Science 336, 1715-1719
5. Nagy, L. G., Riley, R., Tritt, A., Adam, C., Daum, C., Floudas, D., Sun, H., Yadav, J. S., Pangilinan, J., Larsson, K. H., Matsuurа, K., Barry, K., Labutti, K., Kuo, R., Ohm, R. A., Bhattacharya, S. S., Shirouzu, T., Yoshinaga, Y., Martin, F. M., Grigoriev, I. V., and Hibbett, D. S. (2016) Comparative genomics of early-diverging mushroom-forming fungi provides insights into the origins of lignocellulose decay capabilities. Mol. Biol. Evol. 33, 959-970
6. Blanchette, R. A., Burns, T. A., Eerdmans, M. M., and Akhtar, M. (1992) Evaluating isolates of Phanerochaete chrysosporium and Ceriporiopsis subvermispora for use in biological pulping processes. Holzforschung 46, 109-115
7. Blanchette, R. A., Krueger, E. W., Haight, J. E., Akhtar, M., and Akin, D. E. (1997) Cell wall alterations in loblolly pine wood decayed by the white-rot fungus, Ceriporiopsis subvermispora. J. Biotechnol. 53, 203-213
8. Camarero, S., Sarkar, S., Ruiz-Dueñas, F. J., Martínez, M. J., and Martínez, A. T. (1999) Description of a versatile peroxidase involved in the natural degradation of lignin that has both manganese peroxidase and lignin peroxidase substrate interaction sites. J. Biol. Chem. 274, 10324-10330
9. Mester, T., and Field, J. A. (1998) Characterization of a novel manganese peroxidase-lignin peroxidase hybrid isozyme produced by Bjerkandera species strain BOS55 in the absence of manganese. J. Biol. Chem. 273, 15412-15417
Veratryl alcohol radical not released by lignin peroxidase

10. Paszczynski, A., Huynh, V. B., and Crawford, R. (1986) Comparison of ligninase-1 and peroxidase-M2 from the white-rot fungus Phanerochaete chrysosporium. Arch. Biochem. Biophys. 244, 750-765

11. Candeias, L. P., and Harvey, P. J. (1995) Lifetime and reactivity of the veratryl alcohol radical cation. Implications for lignin peroxidase catalysis. J. Biol. Chem. 270, 16745-16748

12. Harvey, P. J., Schoemaker, H. E., and Palmer, J. M. (1986) Veratryl alcohol as a mediator and the role of radical cations in lignin biodegradation by Phanerochaete chrysosporium. FEBS Lett. 195, 242-246

13. Tien, M., Kirk, T. K., Bull, C., and Fee, J. A. (1986) Steady-state and transient-state kinetic studies on the oxidation of 3,4-dimethoxybenzyl alcohol catalyzed by the ligninase of Phanerochaete chrysosporium. J. Biol. Chem. 261, 1687-1693

14. Acebes, S., Ruiz-Dueñas, F. J., Toubes, M., Sáez-Jiménez, V., Pérez-Boada, M., Lucas, M. F., Martínez, A. T., and Guallar, V. (2017) Mapping the long-range electron transfer route in ligninolytic peroxidases. J. Phys. Chem. B 121, 3946-3954

15. Sáez-Jiménez, V., Baratto, M. C., Pogni, R., Rencoret, J., Gutiérrez, A., Santos, J. I., Martínez, A. T., and Ruiz-Dueñas, F. J. (2015) Demonstration of lignin-to-peroxidase direct electron transfer: A transient-state kinetics, directed mutagenesis, EPR, and NMR study. J. Biol. Chem. 290, 23201-23213

16. Sáez-Jiménez, V., Rencoret, J., Rodríguez-Carvajal, M. A., Gutiérrez, A., Ruiz-Dueñas, F. J., and Martínez, A. T. (September 17, 2016) Role of surface tryptophan for peroxidase oxidation of nonphenolic lignin. Biotechnol. Biofuels 9, 198. https://doi.org/10.1186/s13068-016-0615-x

17. Hammel, K. E., Kalyanaraman, B., and Kirk, T. K. (1986) Substrate free radicals are intermediates in ligninase catalysis. Proc. Natl. Acad. Sci. USA 83, 3708-3712

18. Kirk, T. K., Tien, M., Kersten, P. J., Mozuch, M. D., and Kalyanaraman, B. (1986) Ligninase of Phanerochaete chrysosporium. Mechanism of its degradation of the nonphenolic arylglycerol-β-aryl ether substructure of lignin. Biochem. J. 236, 279-287

19. de Jong, E., Field, J. A., and de Bont, J. A. M. (1994) Aryl alcohols in the physiology of ligninolytic fungi. FEMS Microbiol. Rev. 13, 153-188

20. Khindaria, A., Grover, T. A., and Aust, S. D. (1995) Evidence for formation of the veratryl alcohol cation radical by lignin peroxidase. Biochemistry 34, 6020-6025

21. Baciocchi, E., Bietti, M., Francesca Gerini, M., and Lanzalunga, O. (2002) The mediation of veratryl alcohol in oxidations promoted by lignin peroxidase. The lifetime of veratryl alcohol radical cation. Biochem. Biophys. Res. Commun. 293, 832-835

22. Bietti, M., Baciocchi, E., and Steenken, S. (1998) Lifetime, reduction potential and base-induced fragmentation of the veratryl alcohol radical cation in aqueous solution. Pulse radiolysis studies on a ligninase "mediator". J. Phys. Chem. A 102, 7337-7342

23. Bono, J. J., Goulas, P., Boe, J. F., Portet, N., and Seris, J. L. (1990) Effect of Mn(II) on reactions catalyzed by lignin peroxidase from Phanerochaete chrysosporium. Eur. J. Biochem. 192, 189-193

24. Hammel, K. E., Tien, M., Kalyanaraman, B., and Kirk, T. K. (1985) Mechanism of oxidative Cα-Cβ cleavage of a lignin model dimer by Phanerochaete chrysosporium ligninase. Stoichiometry and involvement of free radicals. J. Biol. Chem. 260, 8348-8353

25. Goodwin, D. C., Aust, S. D., and Grover, T. A. (1995) Evidence for veratryl alcohol as a redox mediator in lignin peroxidase-catalyzed oxidation. Biochemistry 34, 5060-5065

26. Koduri, R. S., and Tien, M. (1995) Oxidation of guaiacol by lignin peroxidase: Role of veratryl alcohol. J. Biol. Chem. 270, 22254-22258

27. Tien, M., and Ma, D. (1997) Oxidation of 4-methoxymandelic acid by lignin peroxidase. Mediation by veratryl alcohol. J. Biol. Chem. 272, 8912-8917

28. Hammel, K. E., Jensen, K. A., Mozuch, M. D., Landucci, L. L., Tien, M., and Pease, E. A. (1993) Ligninolysis by a purified lignin peroxidase. J. Biol. Chem. 268, 12274-12281
29. Khindaria, A., Yamazaki, I., and Aust S. D. (1996) Stabilization of the veratryl alcohol cation radical by lignin peroxidase. *Biochemistry* **35**, 6418-6424

30. Drummen, G. P. C., Gadella, B. M., Post, J. A., and Brouwers, J. F. (2004) Mass spectrometric characterization of the oxidation of the fluorescent lipid peroxidation reporter molecule C11-BODIPY 581/591. *Free Radic. Biol. Med.* **36**, 1635-1644

31. Hunt, C. G., Houtman, C. J., Jones, D. C., Kitin, P., Korripally, P., and Hammel, K. E. (2013) Spatial mapping of extracellular oxidant production by a white rot basidiomycete on wood reveals details of ligninolytic mechanism. *Environ. Microbiol.* **15**, 956-966

32. Korripally, P., Hunt, C. G., Houtman, C. J., Jones, D. C., Kitin, P. J., Cullen, D., and Hammel, K. E. (2015) Regulation of gene expression during the onset of ligninolytic oxidation by *Phanerochaete chrysosporium* on spruce wood. *Appl. Environ. Microbiol.* **81**, 7802-7812

33. Dimonie, M. V., Boghina, C. M., Marinescu, N. N., Marinescu, M. M., Cincu, C. I., and Oprescu, C. G. (1982) Inverse suspension polymerization of acrylamide. *Eur. Polym. J.* **18**, 639-645

34. Gorrebeeck, C., Spanoghe, M., Lanens, D., Lemiere, G. L., Dommisse, R. A., Lepoivre, J. A., and Alderweireldt, F. C. (1991) Permeability studies on horse liver alcohol dehydrogenase (HLAD) in polyacrylamide gel beads. *Recl. Trav. Chim. Pays-Bas* **110**, 231-235

35. Kersten, P. J., Kalyanaraman, B., Hammel, K. E., Reinhammar, B., and Kirk, T. K. (1990) Comparison of lignin peroxidase, horseradish peroxidase and laccase in the oxidation of methoxybenzenes. *Biochem. J.* **268**, 475-480

36. Koduri, R. S., Whitwam, R. E., Barr, D., Aust, S. D., and Tien, M. (1996) Oxidation of 1,2,4,5-tetramethoxybenzene by lignin peroxidase of *Phanerochaete chrysosporium*. *Arch. Biochem. Biophys.* **326**, 261-265

37. Branchi, B., Galli, C., and Gentili, P. (2005) Kinetics of oxidation of benzyl alcohols by the dication and radical cation of ABTS. Comparison with laccase-ABTS oxidations: an apparent paradox. *Org. Biomol. Chem.* **3**, 2604-2614

38. Cussler, E. L. (1984) *Diffusion: Mass Transfer in Fluid Systems*, Cambridge University Press, Cambridge

39. Dolman, D., Newell, G. A., and Thurlow, M. D. (1975) A kinetic study of the reaction of horseradish peroxidase with hydrogen peroxide. *Can. J. Biochem.* **53**, 495-501

40. Harvey, P. J., Palmer, J. M., Schoemaker, H. E., Dekker, H. L., and Wever, R. (1989) Pre-steady-state kinetic study on the formation of Compound I and II of ligninase. *Biochim. Biophys. Acta* **994**, 59-63

41. Khindaria, A., Yamazaki, I., and Aust, S. D. (1995) Veratryl alcohol oxidation by lignin peroxidase. *Biochemistry* **34**, 16860-16869

42. Niesner, R., and Heintz, A. (2000) Diffusion coefficients of aromatics in aqueous solution. *J. Chem. Eng. Data* **45**, 1121-1124

43. van Stroe-Biezen, S. A. M., Everaerts, F. M., Janssen, L. J. J., and Tacken, R. A. (1993) Diffusion coefficients of oxygen, hydrogen peroxide and glucose in a hydrogel. *Anal. Chim. Acta* **273**, 553-560

44. Goez, M., and Eckert, G. (1991) Flash CIDNP study of photosensitized electron transfer reactions of methoxybenzenes. *Ber. Bunsenges. Phys. Chem.* **95**, 1179-1186

45. Cai, D. Y., and Tien, M. (1992) Kinetic studies on the formation and decomposition of compound II and compound III. Reactions of lignin peroxidase with H2O2. *J. Biol. Chem.* **267**, 11149-11155

46. Farrell, R. L., Murtagh, K. E., Tien, M., Mozuch, M. D., and Kirk, T. K. (1989) Physical and enzymatic properties of lignin peroxidase isozymes from *Phanerochaete chrysosporium*. *Enzyme Microb. Technol.* **11**, 322-329

47. Recabarren, R., Fuenzalida-Valdivia, I., and Alzate-Morales, J. (2016) Studying the binding mechanisms of veratryl alcohol to *P. chrysosporium* lignin peroxidase: insights from theoretical approaches. *Theor. Chem. Acc.* **135**, 1-12
48. Forney, L. J., Reddy, C. A., Tien, M., and Aust, S. D. (1982) The involvement of hydroxyl radical derived from hydrogen peroxide in lignin degradation by the white rot fungus *Phanerochaete chrysosporium*. *J. Biol. Chem.* 257, 1455-1462

49. Kremer, S. M. and Wood, P. M. (1992) Evidence that cellobiose oxidase from *Phanerochaete chrysosporium* is primarily an Fe(III) reductase: kinetic comparison with neutrophil NADPH oxidase and yeast flavocytochrome B2. *Eur. J. Biochem.* 205, 133-138

50. Hildén, L., Johansson, G., Pettersson, G., Li, J., Ljungquist, P., and Henriksson, G. (2000) Do the extracellular enzymes cellobiose dehydrogenase and manganese peroxidase form a pathway in lignin biodegradation? *FEBS Lett.* 477, 79-83

51. Urano, Y., Higuchi, T., and Hirobe, M. (1996) Substrate-dependent changes of the oxidative *O*-dealkylation mechanism of several chemical and biological oxidizing systems. *J. Chem. Soc. Perkin Trans. 2*, 1169-1173

52. Filley, T. R., Cody, G. D., Goodell, B., Jellison, J., Noser, C., and Ostrofsky, A. (2002) Lignin demethylation and polysaccharide decomposition in spruce sapwood degraded by brown rot fungi. *Org. Geochem.* 33:111-124.

53. Stone, J. E. and Scallan, A. M. (1968) A structural model for the cell wall of water-swollen wood pulp fibers based on their accessibility to macromolecules. *Cellulose Chem. Technol.* 2:343-358.

54. Inman, J. K., and Dintzis, H. M. (1969) Derivatization of cross-linked polyacrylamide beads. Controlled introduction of functional groups for preparation of special-purpose, biochemical adsorbents. *Biochemistry* 8, 4074-4082

55. Tien, M., and Kirk, T. K. (1988) Lignin peroxidase of *Phanerochaete chrysosporium*. *Methods Enzymol.* 161, 238-249

56. Doyle, W. A., and Smith, A. T. (1996) Expression of lignin peroxidase H8 in *Escherichia coli*: Folding and activation of the recombinant enzyme with Ca\(^{2+}\) and haem. *Biochem. J.* 315, 15-19

57. Lakowicz, J. R. (1983) *Principles of Fluorescence Spectroscopy*, Plenum Press, New York

**Footnotes**

This work was supported by grant DE-SC0006929 from the U.S. Department of Energy, Office of Biological and Environmental Research to KEH, CJH, and CGH.

The abbreviations used are: HRP, horseradish peroxidase; LiP, lignin peroxidase; SET, single electron transfer; TMB, 1,2,4,5-tetramethoxybenzene; TMB\(^{\ast}\), 1,2,4,5-tetramethoxybenzene cation radical; VA, veratryl alcohol; VA\(^{\ast}\), veratryl alcohol cation radical.
Table 1

Reaction rate constants used in this study

Reactions to which the values apply are shown in Figure 1.

| Name          | Value               | Notes and references                                    |
|---------------|---------------------|---------------------------------------------------------|
| $k_1$ (LiP/VA)| 580 mM$^{-1}$ s$^{-1}$ | (13)                                                    |
| $k_1$ (HrP/TMB)| 14,100 mM$^{-1}$ s$^{-1}$ | (39)                                                   |
| $k_2$ (LiP/VA)| 90 mM$^{-1}$ s$^{-1}$  | (40)                                                    |
| $k_2$ (HrP/TMB)| 0 mM$^{-1}$ s$^{-1}$   | Pathway inoperative because TMB•• is released quantitatively (35, 36) |
| $k_3$ (LiP)   | 720 mM$^{-1}$ s$^{-1}$ | Calculated from extents of LiP-catalyzed BODIPY bead oxidation without VA |
| $k_3$ (HrP)   | 1200 mM$^{-1}$ s$^{-1}$| Calculated from extents of HrP-catalyzed BODIPY bead oxidation without TMB |
| $k_4$ (LiP/VA)| 330 mM$^{-1}$ s$^{-1}$ | (41)                                                    |
| $k_4$ (HrP/TMB)| > 396 mM$^{-1}$ s$^{-1}$ | $k_6$ assumed to be rate-limiting                       |
| $k_5$ (LiP)   | 0.78 s$^{-1}$       | Calculated from rate of VA oxidation, assuming $k_5 < k_2$, Fig. S2 |
| $k_5$ (HrP)   | 0 mM$^{-1}$ s$^{-1}$ | Pathway inoperative because TMB•• is released quantitatively (35, 36) |
| $k_6$ (LiP/VA)| 10 mM$^{-1}$ s$^{-1}$ | Assumed to be rate-limiting in the pathway that releases VA•• ($k_6 < k_4$). Assumed that 10% of LiP turnovers involving VA proceed via this pathway (23, 40) |
| $k_6$ (HrP/TMB)| 396 mM$^{-1}$ s$^{-1}$ | Measured from rate of TMB oxidation, assuming $k_6 < k_4$, Fig. S1 |
| $k_7$ (LiP/VA)| 17 s$^{-1}$         | (11, 21, 22)                                           |
| $k_7$ (HrP/TMB)| 0.03 s$^{-1}$       | Measured, Fig. S1                                       |
| $k_8$         | $10^6$ mM$^{-1}$ s$^{-1}$ | Diffusion-limited                                      |
| $k_9$         | 600 mM$^{-1}$ s$^{-1}$ | Set to give best fit of model to data                 |
FIGURE 1: Reactions used for both the reaction-diffusion model and the solution-phase model. The named species are: $E_0$, ferric resting state peroxidase; $O=E_i$, two-electron-oxidized peroxidase compound I; $HO-E_{ii}$, one-electron-oxidized peroxidase compound II; $HO-E_{ii}-M^{+\cdot}$, peroxidase compound II associated with mediator cation radical; $M$, mediator (TMB or VA); $M^{+\cdot}$, mediator cation radical; $A$, two-electron-oxidized product derived from mediator (2,5-dimethoxybenzoquinone or veratraldehyde); $R$, native red BODIPY dye; $G$, oxidized green BODIPY dye; $H_2O_2$; and $HO_2^\cdot$. Values for the rate constants are given in Table 1.
FIGURE 2. Confocal fluorescence images of a BODIPY-functionalized bead in a solution of fluorescein-labeled ovalbumin. The red channel, detecting native nonoxidized BODIPY, is shown at the top and the green channel, detecting fluorescein, is shown at the bottom. The green channel is shown in false color (cyan) to avoid confusion with the green fluorescence of oxidized BODIPY that appears in the following figures. The small boxes in the images on the left are 10 × 10 µm in size, and are shown magnified on the right.
FIGURE 3. Green fluorescence intensity of BODIPY-functionalized beads when immersed in a solution of fluorescein-labeled ovalbumin. The intensity within the beads is expressed as a fraction of the intensity in the surrounding solution. The data points are from image analyses of nine beads.
FIGURE 4. Imaging data for beads treated with peroxidases, TMB, and H$_2$O$_2$. A. Confocal fluorescence images of beads treated with various levels of H$_2$O$_2$, using excess TMB (4 mM) and LiP. The image sequence, from left to right, is for 0.00, 0.05, 0.10, and 0.40 mM H$_2$O$_2$. The images have been adjusted to represent constant excitation laser power. The scale bar is 10 µm. B. Experimental dye oxidation profiles (data points for triplicate beads) and modeled dye oxidation profiles (lines) for beads treated with increasing levels of TMB, using excess H$_2$O$_2$ (286 µM) and HrP. Blue, 1.4 µM TMB; red, 2.8 µM TMB; green, 4.3 µM TMB. C. Comparison of experimental data (closed circles) and reaction-diffusion model (open circles) for the depth into the beads at which 50% oxidation occurred after treatment with the HrP/H$_2$O$_2$/TMB system. Results are shown as a function of oxidant dosage, i.e., the amount of TMB supplied. Error bars on the experimental values indicate 95% confidence intervals. The colored arrows indicate the correspondence to the data in plot B.
FIGURE 5. Imaging data for beads treated with LiP, VA, and H$_2$O$_2$. A. Confocal fluorescence images of beads exposed to the following treatments, from left to right: H$_2$O$_2$ alone (0.8 mM); LiP and H$_2$O$_2$ (0.8 mM); LiP, VA (12 mM) and H$_2$O$_2$ (0.8 mM). The images have been adjusted to represent constant excitation laser power. The scale bar is 10 µm. B. Quantitative analyses of eight images each of beads exposed to LiP and 0.8 mM H$_2$O$_2$, with and without 12 mM VA.
FIGURE 6. Imaging data for beads treated with LiP and VA using increasing doses of H$_2$O$_2$. A. Confocal fluorescence images of beads exposed to 12 mM VA and LiP, using the following H$_2$O$_2$ concentrations, from left to right: 0.5, 1.0, 2.0 and 4.0 mM. The images have been adjusted to represent constant excitation laser power. The scale bar is 10 µm. B. Experimental dye oxidation profiles (data points for eight replicate beads) and modeled dye oxidation profiles (lines) for beads treated with increasing levels of H$_2$O$_2$, using excess VA (12 mM) and LiP. Yellow, 0.5 mM H$_2$O$_2$; blue, 1.0 mM H$_2$O$_2$; red, 2.0 mM H$_2$O$_2$; green, 4.0 mM H$_2$O$_2$. C. Comparison of experimental data (closed circles) and reaction-diffusion model (open circles) for the depth into the beads at which 50% oxidation occurred after treatment with the LiP/H$_2$O$_2}$/VA system. Results are shown as a function of oxidant dosage, i.e., the amount of H$_2$O$_2$ supplied. Error bars on the experimental values indicate 95% confidence intervals. The colored arrows indicate the correspondence to the data in plot B.
Fungal lignin peroxidase does not produce the veratryl alcohol cation radical as a diffusible ligninolytic oxidant

Carl J. Houtman, Eranda Maligaspe, Christopher G. Hunt, Elena Fernández-Fueyo, Angel T. Martínez, and Kenneth E. Hammel

CONTENTS

Figure S1. Spectrophotometric assessment of TMB oxidation by HrP S2
Figure S2. Spectrophotometric assessment of VA oxidation by LiP S3
Table S1. Stoichiometries of veratraldehyde production by LiP at increasing levels of H₂O₂ S4
Figure S3. Alignments of LiP sequences for the P. chrysosporium genome S5
Figure S4. Predicted models of the surface environment near the catalytic tryptophan in P. chrysosporium LiPs S7
Data Set S1. Computer code for the reaction-diffusion model and the solution-phase model S8
Fig. S1. Comparison of experimentally determined absorbance of TMB cation radical, produced by HrP from TMB, with a prediction of the reaction in solution using the rate constants determined for this work.

\[ K_6 = 396 \text{ mM}^{-1} \text{ sec}^{-1} \]
\[ K_7 = 0.03 \text{ sec}^{-1} \]
Fig. S2: Comparison of experimentally determined absorbance of veratraldehyde, produced by LiP from VA, with a prediction of the reaction in solution using the rate constant determined for this work.

$K_5 = 0.777 \text{ sec}^{-1}$
Table S1
Stoichiometries of veratraldehyde production at increasing levels of H$_2$O$_2$ in reactions containing LiP, veratryl alcohol, and BODIPY dye immobilized in porous beads

Reactions were conducted as described under “Experimental Procedures,” using 12 mM veratryl alcohol.

| H$_2$O$_2$ supplied (mM) | Veratraldehyde produced (mM) | Yield (proportion of H$_2$O$_2$ converted) |
|-------------------------|------------------------------|------------------------------------------|
| 0.50                    | 0.42                         | 0.84                                     |
| 0.80                    | 0.70                         | 0.88                                     |
| 1.00                    | 0.87                         | 0.87                                     |
| 2.00                    | 1.57                         | 0.79                                     |
| 4.00                    | 2.52                         | 0.63                                     |
Fig. S3. LiP sequences from the *Phanerochaete chrysosporium* genome (RP-78 v2.2)

**Residues** (11) highlighted in purple are predicted to interact with VA in the W171 environment according to Acebes et al. [L167, V170, C249, E250, W251, S253, E254, N257, K260, D263, D264].

Residues (6) marked ^ are predicted to interact with VA in the W171 environment according to Recabarren et al. [F164, E168, D263, D264, F267, I268].

Residues (20) marked # are predicted to make up the entire W171 surface environment in the models shown in Fig. S4. [N127, E163, D165, E166, L167, E168, R246, C249, E250, S253, N257, Q258, S259, K260, D263, D264, Q266, F267, L270, A271].

Differences that could affect the binding vs. release of VA^+• include two acidic residues. E163 in LiPA changes to G in LiPD and to Q in LiPG/H/I. D263 in LiPA changes to S in LiPB/C/G/H/I/J and to E in LiPD.

Acebes, S., Ruiz-Dueñas, F. J., Toubes, M., Sáez-Jiménez, V., Pérez-Boada, M., Lucas, M. F., Martínez, A. T., and Guallar, V. (2017) Mapping the long-range electron transfer route in ligninolytic peroxidases. *J. Phys. Chem. B* **121**, 3946-3954

Recabarren, R., Fuenzalida-Valdivia, I., and Alzate-Morales, J. (2016) Studying the binding mechanisms of veratryl alcohol to *P. chrysosporium* lignin peroxidase: insights from theoretical approaches. *Theor. Chem. Acc.* **135**, 1-12
Fig. S4. Predicted models of the surface environment near the catalytic tryptophan in the ten P. chrysosporium LiPs.
DATA SET S1. Computer code for the reaction-diffusion model and the solution-state model

# unsteady-state reaction in a solution
# The species:
#  Eo = reduced enzyme
#  Ei = enzyme I
#  Eii = enzyme II
#  Eiis = enzyme II associated with substrate
#  Vo = veratryl alcohol
#  Vr = oxidized veratryl alcohol (cation radical)
#  Va = veratryl aldehyde
#  P = peroxide
#  Pr = super oxide

ch 12/14/15

import the modules from time import time import numpy as np import math from scipy import integrate import matplotlib.pyplot as plt

# define the parameters
Ce = 0.003375 # enzyme concentration (mM) Cv = 12.0 # VA concentration (mM) Cp = 4.0 # peroxide concentration (mM) Cd = 0.0 # dye concentration (mM) K1 = 580.0 # rxn Eo + P --> Ei + H2O (1/mM s) K2 = 90.0 # rxn Ei + Vo --> Eiis (1/mM s) K3 = 100.0 # rxn Ei + D --> Eii + Dox (1/mM s) K4 = 10.0 # rxn Ei + Vo --> Eii + Vr (1/mM s) K5 = 0.777464424 # rxn Eiis --> Eo + Va (1/s) K6 = 10.0 # rxn Eii + Vo --> Eo + Vr (1/mM s) K7 = 17.0 # rxn Vr -->  + O2 --> Va + HOO. (1/s) (assume O2 is constant) K8 = 1000000.0 # rxn HOO. + HOO. --> P + O2 (1/mM s)
suffix = "LiP_full"

# Set the time range and number of steps
t_start = 0.0   # sec t_final = 2000.0
num_steps = 100000 store_factor = 100 # number of steps between storing num_store = num_steps/store_factor + 1 delta_t = float((t_final - t_start)/(num_steps-1))

# define the reaction equations
def RnxRate(t, y):
    # Assign some variables for convenience of notation
    Ei, Eii, Eiis, Eo, Vo, Vr, Va, D, P, Pr = y.T
dEi = K1*Eo*P - (K2+K4)*Ei*Vo - K3*Ei*D # dEi/dt
dEii = K3*Ei*Vo + K4*Ei*Vo - K6*Eii*Vo # dEii/dt
dEiis = K2*Ei*Vo - K5*Eiis # dEiis/dt
dEo = -K1*Eo*P + K5*Eiis + K6*Eii*Vo # dEo/dt
dVo = -(K2+K4)*Ei*Vo - K6*Eii*Vo # dVo/dt
dVr = K4*Ei*Vo + K6*Eii*Vo - K7*Vr # dVr/dt
dVa = K7*Vr + K5*Eiis # dVa/dt
dD = -K3*Ei*D # dD/dt
dP = -K1*Eo*P + K8*Pr*Pr # dP/dt
dPr = K7*Vr - K8*Pr*Pr # dPr/dt

return np.array((dEi, dEii, dEiis, dEo, dVo, dVr, dVa, dD, dP, dPr))

# define the Jacobian
def RnxJac(t, y):
# Assign some variables for convenience of notation
Ei, Eii, Eiis, Eo, Vo, Vr, Va, D, P, Pr = y.T
jEi = [-(K2+K4)*Vo-K3*D, 0.0, 0.0, K1*P, -(K2+K4)*Ei, 0.0, 0.0, -K3*Ei, K1*Eo, 0.0]
jEii = [K4*Vo+K3*D, -K6*Vo, 0.0, 0.0, K4*Ei-K6*Eii, 0.0, 0.0, K3*Ei, 0.0, 0.0]
jEiis = [K2*Vo, 0.0, -K5, 0.0, K2*Ei, 0.0, 0.0, 0.0, 0.0, 0.0]
jEo = [0.0, K6*Vo, K5, -K1*P, K6*Eii, 0.0, 0.0, 0.0, 0.0, 0.0]
jVo = [-(K2+K4)*Vo, -K6*Vo, 0.0, 0.0, -(K2+K4)*Ei-K6*Eii, 0.0, 0.0, 0.0, 0.0, 0.0]
jVr = [K4*Vo, K6*Vo, 0.0, 0.0, K4*Ei+K6*Eii, -K7, 0.0, 0.0, 0.0, 0.0]
jVa = [0.0, 0.0, K5, 0.0, 0.0, K7, 0.0, 0.0, 0.0, 0.0]
jD = [-K3*D, 0.0, 0.0, 0.0, 0.0, 0.0, -K3*Ei, 0.0, 0.0, 0.0]
jP = [0.0, 0.0, 0.0, -K1*P, 0.0, 0.0, 0.0, -K1*Eo, 2.0*K8*Pr]
jPr = [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -2.0*K8*Pr]
return np.array((jEi, jEii, jEiis, jEo, jVo, jVr, jVa, jD, jP, jPr))

# initialize the variables
Ei = np.zeros(num_store)
Eii = np.zeros(num_store)
Eiis = np.zeros(num_store)
Eo = np.zeros(num_store)
Vo = np.zeros(num_store)
Vr = np.zeros(num_store)
Va = np.zeros(num_store)
D = np.zeros(num_store)
P = np.zeros(num_store)
Pr = np.zeros(num_store)
t = np.zeros(num_store)

# set initial conditions using some algebra to eliminate the initial stiffness
Ei[0] = Ce/(1+K2*Cv/K5+K4/K6+(K2+K4)*Cv/(K1*Cp))
Eii[0] = K4*Ei[0]/K6
Eiis[0] = K2*Ei[0]*Cv/K5
Eo[0] = (K2+K4)*Ei[0]*Cv/(K1*Cp)
Vo[0] = Cv
Vr[0] = (K4*Ei[0]+K6*Eii[0])*Cv/K7
Va[0] = 0.0
D[0] = Cd
P[0] = Cp
Pr[0] = math.sqrt(K7*Vr[0]/K8)
t[0] = t_start

# use 'VODE' with a full Jacobian
r = integrate.ode(RxnRate, RxnJac).set_integrator('vode', with_jacobian=True)

# Set initial condition(s): for integrated variables and time
r.set_initial_value((Ei[0], Eii[0], Eiis[0], Eo[0], Vo[0], Vr[0], Va[0], D[0], P[0], Pr[0]), t[0])

myTime = time() # get the initial clock time

# solve the equations
k = 1
j = 1
while r.successful() and k < num_steps:
    r.integrate(r.t + delta_t)
    # Store the results for later
    if k % store_factor == 0:
        t[j] = r.t
        j += 1
    k += 1
Ei[j], Eii[j], Eiis[j], Eo[j], Vo[j], Vr[j], Va[j], D[j], P[j], Pr[j] = r.y.T
j += 1

k += 1

# store the last line
r.t = t[j]
Ei[j], Eii[j], Eiis[j], Eo[j], Vo[j], Vr[j], Va[j], D[j], P[j], Pr[j] = r.y.T

# print a notification of completion
print "total time simulated = ", t_final
print "elapsed time = ", time()-myTime, " seconds"

# store the results in a csv file
try:
    myFile = open("solution_model_"+suffix+".csv","w")
except IOError as err:
    print("can't open file")
else:
    # put a header on the file
    myFile.write("total time simulated = {} seconds\n".format(t_final))
    myFile.write("enzyme concentration (mM) = {}\n".format(Ce))
    myFile.write("VA concentration (mM) = {}\n".format(Cv))
    myFile.write("peroxide concentration (mM) = {}\n".format(Cp))
    myFile.write("dye concentration (mM) = {}\n".format(Cd))
    myFile.write("K1 = {} rxn Eo + P --> Ei + H2O (1/mM s)\n".format(K1))
    myFile.write("K2 = {} rxn Ei + Vo --> Eiis  (1/mM s)\n".format(K2))
    myFile.write("K3 = {} rxn Ei + D  --> Eii + Dox (1/mM s)\n".format(K3))
    myFile.write("K4 = {} rxn Ei + Vo --> Eii + Vr  (1/mM s)\n".format(K4))
    myFile.write("K5 = {} rxn Eiis  --> Eo + Va (1/s)\n".format(K5))
    myFile.write("K6 = {} rxn Eii + Vo  --> Eo + Vr (1/mM s)\n".format(K6))
    myFile.write("K7 = {} rxn Vr + O2  --> Va + HOO. (1/s) (assume O2 is constant)\n".format(K7))
    myFile.write("K8 = {} rxn HOO. + HOO.  --> P + O2 (1/mM s)\n".format(K8))
    myFile.write("Time(sec),EI,EII,EII+s,Eo, Veratryl Alcohol, Cation Radical, Veratryl Aldehyde, Dye, Peroxide, Peroxyl Radical\n")
    for j in range(0, num_store):
        myFile.write("{}\n".format(t[j], Ei[j], Eii[j], Eiis[j], Eo[j], Vo[j], Vr[j], Va[j], D[j], P[j], Pr[j]))
    myFile.close()

# Plot the trajectories
fig = plt.figure()
ax1 = fig.add_subplot(311)
ax1.plot(t, Va, 'b-', label='Val')
ax1.legend(loc='best')
ax1.set_xlabel('Time [sec]')
ax1.set_ylabel('Concentration [mmol/L]')
ax1.set_xlim([0, t_final])
ax1.set_ylim([0, np.max(Va)*1.1])
ax1.grid('on')

ax2 = fig.add_subplot(312)
ax2.plot(t, Vr, 'b-', label='Vrad')
ax2.legend(loc='best')
ax2.set_xlabel('Time [sec]')
ax2.set_ylabel('Concentration [mmol/L]')
ax2.set_xlim([0, t_final])
ax2.set_ylim([0, np.max(Vr)*1.1])
ax2.grid('on')

ax3 = fig.add_subplot(313)
ax3.plot(t, Eo, 'r-', label='Eo')
ax3.plot(t, Ei, 'b-', label='Ei')
ax3.plot(t, Eii, 'g-', label='Eii')
ax3.plot(t, Eiis, 'k-', label='Eiis')
ax3.legend(loc='best')
ax3.set_xlabel('Time [sec]')
ax3.set_ylabel('Concentration [mmol/L]')
ax3.set_xlim([0, t_final])
ax3.set_ylim([0, Ce*1.1])
ax3.grid('on')

fig.show()
unsteady-state reaction in a solution

The species:
- \(E_0\) = reduced enzyme
- \(E_1\) = enzyme I
- \(E_{11}\) = enzyme II
- \(E_{111}\) = enzyme II associated with substrate
- \(V_0\) = veratryl alcohol
- \(V_r\) = oxidized veratryl alcohol (cation radical)
- \(V_a\) = veratryl aldehyde
- \(P\) = peroxide
- \(P_r\) = super oxide

ch 12/14/15

import the modules
from time import time
import numpy as np
import math
from scipy import integrate
import matplotlib.pyplot as plt

define the parameters
\(C_e = 0.00031746\) # enzyme concentration (mM)
\(C_v = 0.000714\) # TMB concentration (mM)
\(C_v^* = 2.0\) # of doses
\(C_p = 0.285714286\) # peroxide concentration (mM)
\(C_d = 0.0\) # dye concentration (mM)
\(K_1 = 580.0\) # rxn \(E_0 + P \rightarrow E_1 + H_2O\) (1/mM s)
\(K_2 = 0.01\) # rxn \(E_1 + V_0 \rightarrow E_{11}\) (1/mM s)
\(K_3 = 100.0\) # rxn \(E_1 + D \rightarrow E_{11} + Dox\) (1/mM s)
\(K_4 = 1000.0\) # rxn \(E_1 + V_0 \rightarrow E_{11} + V_r\) (1/mM s)
\(K_5 = 0.01\) # rxn \(E_{11} \rightarrow E_0 + V_a\) (1/s)
\(K_6 = 396.0\) # rxn \(E_{11} + V_0 \rightarrow E_0 + V_r\) (1/mM s)
\(K_7 = 0.029548258748605\) # rxn \(V_r \rightarrow O_2 \rightarrow V_a + HOO\). (1/s) (assume \(O_2\) is constant)
\(K_8 = 1000000.0\) # rxn \(HOO. + HOO. \rightarrow P + O_2\) (1/mM s)
suffix = "HRP_full_false"

Set the time range and number of steps
t_start = 0.0 # sec
t_final = 300.0
num_steps = 100000
store_factor = 100 # number of steps between storing
num_store = num_steps/store_factor + 1
delta_t = float((t_final - t_start)/(num_steps-1))

def the reaction equations
def RxnRate(t, y):
    # Assign some variables for convenience of notation
    \(E_1\), \(E_{11}\), \(E_{111}\), \(E_0\), \(V_0\), \(V_r\), \(V_a\), \(D\), \(P\), \(P_r\) = y.T
    dEi = K1*E0*P - (K2+K4)*E1*Vo - K3*Ei*D # dEi/dt
    dEii = K3*Ei*D + K4*E1*Vo - K6*E11*Vo # dEii/dt
    dEiis = K2*E1*Vo - K5*E11s # dEiis/dt
    dEo = -K1*E0*P + K5*E11s # dE0/dt
    dVo = -K4*E1*Vo - K6*E11*Vo # dVo/dt
    dVr = K4*E1*Vo + K6*E11*Vo - K7*Vr # dVr/dt
    dVa = K7*Vr + K5*E11 # dVa/dt
    dD = -K3*Ei*D # dD/dt
    dP = -K1*E0*P + K8*Pr*Pr # dP/dt
    dPr = K7*Vr - K8*Pr*Pr # dPr/dt

    return np.array((dEi, dEii, dEiis, dEo, dVo, dVr, dVa, dD, dP, dPr))

def RxnJac(t, y):
    # Assign some variables for convenience of notation
    \(E_1\), \(E_{11}\), \(E_{111}\), \(E_0\), \(V_0\), \(V_r\), \(V_a\), \(D\), \(P\), \(P_r\) = y.T

    # Assign some variables for convenience of notation
    \(E_1\), \(E_{11}\), \(E_{111}\), \(E_0\), \(V_0\), \(V_r\), \(V_a\), \(D\), \(P\), \(P_r\) = y.T

\[
\begin{align*}
\text{jEi} &= \begin{pmatrix} -(K2+K4) \cdot \text{Vo} - K3 \cdot \text{D}, & 0.0, & 0.0, & K1 \cdot \text{P}, & -(K2+K4) \cdot \text{Ei}, & 0.0, & 0.0, & -K3 \cdot \text{Ei}, & K1 \cdot \text{Eo}, & 0.0 \end{pmatrix} \\
\text{jEii} &= \begin{pmatrix} K4 \cdot \text{Vo} + K3 \cdot \text{D}, & -K6 \cdot \text{Vo}, & 0.0, & 0.0, & K4 \cdot \text{Ei} - K6 \cdot \text{Eii}, & 0.0, & 0.0, & K3 \cdot \text{Ei}, & 0.0, & 0.0 \end{pmatrix} \\
\text{jEiis} &= \begin{pmatrix} K2 \cdot \text{Vo}, & 0.0, & -K5, & 0.0, & K2 \cdot \text{Ei}, & 0.0, & 0.0, & 0.0, & 0.0 \end{pmatrix} \\
\text{jEo} &= \begin{pmatrix} 0.0, & K6 \cdot \text{Vo}, & K5, & -K1 \cdot \text{P}, & K6 \cdot \text{Eii}, & 0.0, & 0.0, & 0.0, & -K1 \cdot \text{Eo}, & 0.0 \end{pmatrix} \\
\text{jVo} &= \begin{pmatrix} -(K2+K4) \cdot \text{Vo}, & -K6 \cdot \text{Vo}, & 0.0, & 0.0, & -(K2+K4) \cdot \text{Ei} - K6 \cdot \text{Eii}, & 0.0, & 0.0, & 0.0, & 0.0, & 0.0 \end{pmatrix} \\
\text{jVr} &= \begin{pmatrix} K4 \cdot \text{Vo}, & K6 \cdot \text{Vo}, & 0.0, & 0.0, & K4 \cdot \text{Ei} + K6 \cdot \text{Eii}, & -K7, & 0.0, & 0.0, & 0.0, & 0.0 \end{pmatrix} \\
\text{jVa} &= \begin{pmatrix} 0.0, & 0.0, & K5, & 0.0, & 0.0, & K7, & 0.0, & 0.0, & 0.0, & 0.0 \end{pmatrix} \\
\text{jD} &= \begin{pmatrix} -K3 \cdot \text{D}, & 0.0, & 0.0, & 0.0, & 0.0, & 0.0, & 0.0, & -K3 \cdot \text{Ei}, & 0.0, & 0.0 \end{pmatrix} \\
\text{jP} &= \begin{pmatrix} 0.0, & 0.0, & 0.0, & -K1 \cdot \text{P}, & 0.0, & 0.0, & 0.0, & -K1 \cdot \text{Eo}, & 0.0 \end{pmatrix} \\
\text{jPr} &= \begin{pmatrix} 0.0, & 0.0, & 0.0, & 0.0, & 0.0, & 0.0, & 0.0, & 0.0 \end{pmatrix}
\end{align*}
\]

\[
\text{return } \text{np.array}((\text{jEi, jEii, jEiis, jEo, jVo, jVr, jVa, jD, jP, jPr}))
\]

```python
# initialize the variables
Ei = np.zeros(num_store)
Eii = np.zeros(num_store)
Eiis = np.zeros(num_store)
Eo = np.zeros(num_store)
Vo = np.zeros(num_store)
Vr = np.zeros(num_store)
Va = np.zeros(num_store)
D = np.zeros(num_store)
P = np.zeros(num_store)
Pr = np.zeros(num_store)
t = np.zeros(num_store)

# set initial conditions using some algebra to eliminate the initial stiffness
Ei[0] = Ce/(1+K2*Kv*K5/K6+(K2+K4)*Cv/(K1*Cp))
Eii[0] = K4*Ei[0]/K6
Eiis[0] = K2*Ei[0]*Cv/K5
Eo[0] = (K2+K4)*Ei[0]*Cv/(K1*Cp)
Vo[0] = Cv
Vr[0] = 0.0
Va[0] = 0.0
D[0] = Cd
P[0] = Cp
Pr[0] = math.sqrt(K7*Vr[0]/K8)
t[0] = t_start

# use 'VODE' with a full Jacobian
r = integrate.ode(RxnRate, RxnJac).set_integrator('vode', with_jacobian=True)

# Solve the equations
k = 1
j = 1
while r.successful() and k < num_steps:
    r.integrate(r.t + delta_t)
    # Store the results for later
    if k % store_factor == 0:
        t[j] = r.t
        Ei[j], Eii[j], Eiis[j], Eo[j], Vo[j], Vr[j], Va[j], D[j], P[j], Pr[j] = r.y.T
        j += 1
```

S13
k += 1

# store the last line
t[j] = r.t
Ei[j], Eii[j], Eiis[j], Eo[j], Vo[j], Vr[j], Va[j], D[j], P[j], Pr[j] = r.y.T

# print a notification of completion
print "total time simulated = ", t_final
print "elapsed time = ", time()-myTime, " seconds"

# store the results in a csv file
try:
    myFile = open("solution_model_"+suffix+".csv","w")
except IOError as err:
    print("can't open file")
else:
    # put a header on the file
    myFile.write( "total time simulated = {} seconds 
".format(t_final))
    myFile.write("enzyme concentration (mM) = {}\n".format(CE))
    myFile.write("VA concentration (mM) = {}\n".format(Cv))
    myFile.write("peroxide concentration (mM) = {}\n".format(Cp))
    myFile.write("dye concentration (mM) = {}\n".format(Cd))
    myFile.write("K1 = {} rxn Eo + P --> Ei + H2O (1/mM s) \n".format(K1))
    myFile.write("K2 = {} rxn Ei + Vo --> Eiis  (1/mM s) \n".format(K2))
    myFile.write("K3 = {} rxn Ei + D  --> Eii + Dox (1/mM s) \n".format(K3))
    myFile.write("K4 = {} rxn Ei + Vo --> Eii + Vr  (1/mM s) \n".format(K4))
    myFile.write("K5 = {} rxn Eiis  --> Eo + Va (1/s) \n".format(K5))
    myFile.write("K6 = {} rxn Eii + Vo  --> Eo + Vr (1/mM s) \n".format(K6))
    myFile.write("K7 = {} rxn Vo + O2  --> Va + HOO. (1/s) (assume O2 is constant) \n".format(K7))
    myFile.write("K8 = {} rxn HOO. + HOO. --> P + O2 (1/mM s) \n".format(K8))
    myFile.write("Time(sec),EI,EII,EII+s,Eo, Veratryl Alcohol, Cation Radical, Veratryl Aldehyde, Dye, Peroxide, Peroxyl Radical\n")
    for j in range(0, num_store):
        myFile.write("{},{}
".format(t[j], Ei[j], Eii[j], Eiis[j], Eo[j], Vo[j], Vr[j], Va[j], D[j], P[j], Pr[j]))

myFile.close()

# Plot the trajectories
fig = plt.figure()
ax1 = fig.subplot(311)
ax1.plot(t, P, 'g-', label='P')
ax1.legend(loc='best')
ax1.set_xlabel('Time [sec]')
ax1.set_ylabel('Concentration [mmol/L]')
ax1.set_xlim([0, t_final])
ax1.set_ylim([0, np.max(P)*1.1])
ax1.grid('on')

ax2 = fig.subplot(312)
ax2.plot(t, Vr, 'b-', label='Vrad')
ax2.plot(t, Va, 'g-', label='Val')
ax2.plot(t, Vo, 'r-', label='Vo')
ax2.legend(loc='best')
ax2.set_xlabel('Time [sec]')
ax2.set_ylabel('Concentration [mmol/L]')
ax2.set_xlim([0, t_final])
ax2.set_ylim([0, np.max(Vo)*1.1])
ax2.grid('on')

ax3 = fig.subplot(313)
ax3.plot(t, Eo, 'r-', label='Eo')
ax3.plot(t, Ei, 'b-', label='Ei')
ax3.plot(t, Eiis, 'k-', label='Eiis')
ax3.plot(t, Eii, 'g-', label='Eii')
ax3.legend(loc='best')
ax3.set_xlabel('Time [sec]')
ax3.set_ylabel('Concentration [mmol/L]')
ax3.set_xlim([0, t_final])
ax3.set_ylim([0, Ce*1.1])
ax3.grid('on')

fig.show()
**Reaction of LiP, Peroxide and VA in the beads**

unsteady-state diffusion in a sphere

The species:

- $E_0 =$ reduced enzyme
- $E_1 =$ enzyme I
- $E_{II} =$ enzyme II
- $E_{II} =$ enzyme II associated with substrate
- $V_0 =$ veratryl alcohol
- $V_r =$ oxidized veratryl alcohol (cation radical)
- $V_a =$ veratryl aldehyde
- $D_g =$ oxidized dye (green)
- $D_r =$ reduced dye (red)
- $P =$ peroxide
- $P_r =$ super oxide

ch 5/18/15

import the modules

```python
from fipy import CellVariable, DiffusionTerm, Viewer, TransientTerm, Variable
from fipy.meshes.sphericalUniformGrid1D import SphericalUniformGrid1D
from time import time
import numpy as npy
from scipy import interp
from PIL import Image
```

define the parameters

```python
D_p = 5.0E-6 # peroxide diffusion coefficient (cm^2/sec)
R = 35.0E-4 # radius (cm)
C_e = 0.003375 # enzyme concentration (mM)
C_d = 0.5 # dye concentration (mM)
C_v = 12.0 # VA concentration (mM)
C_p = 0.4226 # peroxide concentration (mM) (note: 0.5=0.4226 0.8=0.6981 1.0=0.8726
2.0=1.572 4.0=2.516)
K_1 = 580.0 # rxn $E_0 + P \rightarrow E_1 + H_2O$ (1/mM s) (tien)
K_2 = 90.0 # rxn $E_1 + V_0 \rightarrow E_{IIis}$ (1/mM s) (harvey)
K_3 = 720.0 # rxn $E_1 + D \rightarrow E_{IIis} + Dox$ (1/mM s) (WAG)
K_4 = 10.0 # rxn $E_1 + V_0 \rightarrow E_{IIis} + V_r$ (1/mM s)
K_5 = 0.777464424 # rxn $E_{IIis} \rightarrow E_0 + V_a$ (1/s) (fit compare to Tien's 0.8)
K_6 = 10.0 # rxn $E_{IIis} + V_0 \rightarrow E_0 + V_r$ (1/mM s)
K_7 = 17.0 # rxn $V_r \rightarrow O_2 \rightarrow V_a + HOO.$ (1/s) (Stenken)
K_8 = 1000000.0 # rxn $HOO. + HOO.$ --$ P + O_2$ (1/mM s)
K_9 = 0.0 # rxn radical + D --$ + Dox$ (1/mM s) (from the HRP results)
dyeR = 100.0 # ratio of red to green bodipy reaction rate with radical
D_v = 0.62 # ratio of VA diffusion to H2O2
D_vr = 0.62 # ratio of VA cation radical diffusion to H2O2
proteinProfile = "noVA" # set the profile "orginal", "40K", "noVA" or "excluded"
```

`solution parameters`

```python
myViewer = False # if True a plot of concentration vs position is shown and updated
steps = 2000000 # number of time steps
printInterval = 10000 # number of step between prints
nx = 100 # number of nodes
dx = 1.0 / nx
dt = 0.0005
```

define a functions to store the dye results

```python
def storeResults():
    try:
        myFile = open("model_"+suffix+".csv","w")
        except IOError as err:
            print("can't open file")
        else:
            # put a header on the file
            myFile.write("total time simulated = {0} seconds
            .format(simTime*R**2/Dp))
            myFile.write("peroxide diffusion coefficient (cm^2/sec) = {0}
            .format(Dp))
            myFile.write("radius (cm) = {0}".format(R))
            myFile.write("enzyme concentration (mM) = {0}
            .format(Ce))
```
myFile.write("VA concentration (mM) = {}
")
myFile.write("peroxide concentration (mM) = {}
")
myFile.write("dye concentration (mM) = {} 
")
myFile.write("K1 = {} rxn Eo + P --> Ei + H2O (1/mM s)
")
myFile.write("K2 = {} rxn Ei + Vo --> Eii (1/mM s)
")
myFile.write("K3 = {} rxn Ei + D --> Eii + Dox (1/mM s)
")
myFile.write("K4 = {} rxn Ei + Vo --> Eii + Vr (1/mM s)
")
myFile.write("K5 = {} rxn Eii + Va --> Eo + Va + H2O (1/s)
")
myFile.write("K6 = {} rxn Eii + Vo --> Eo + Vr (1/mM s)
")
myFile.write("K7 = {} rxn Vr + O2 --> Va + HOO. (1/s)
")
myFile.write("K8 = {} rxn HOO. + HOO. --> P + O2 (1/mM s)
")
myFile.write("K9 = {} rxn radical + D  --> + Dox (1/mM s) 
")
myFile.write("proteinProfile = {} updated 
")
myFile.write("final profile
")
for j in range(0, nx):
    myFile.write("{},{},{},{},{},{},{},{},{},{},{} 
")
myFile.close()

# dimensionless rate constants
Tm1 = K1*R**2/Dp
Tm2 = K2*R**2/Dp
Tm3 = K3*R**2/Dp
Tm4 = K4*R**2/Dp
Tm5 = K5*R**2/Dp
Tm6 = K6*R**2/Dp
Tm7 = K7*R**2/Dp
Tm8 = K8*R**2/Dp
Tm9 = K9*R**2/Dp

# rate of veratyl alcohol and peroxide loss
li = Ce/(1.0+Tm2*Cv/Tm5 + Tm4/Tm6 + (Tm2+Tm4)*Cv/(Tm1*Cp)) #initial enzyme conc.
lii = Tm4*li/Tm6
liis = Tm2*li*Cv/Tm5
liis = Tm2*li*Cv/Tm5

Vri = (Tm4*li+Tm6*lii)*Cv/Tm7
Pri = npy.sqrt(Tm7*Vri/Tm8)

dpdt = -Tm1*lo*Cp #+ Tm8*Pri*Pri  # rate of change of peroxide in solution (assume Va excess)
dvdt = -((Tm2 + Tm4)*li + Tm6*lii)*Cv # rate of change of vertryl alcohol in solution (assume Va excess)

# build a 1D mesh
dim = SphericalUniformGrid1D(nx=nx, dx=dx)

# initialize the cell variables
Eo = CellVariable(mesh=dim, name = "Eo", value = 0.0) # <--- this is set below
Ei = CellVariable(mesh=dim, name = "Ei", value = 0.0) # <--- this is set below
Eii = CellVariable(mesh=dim, name = "Eii", value = 0.0) # <--- this is set below
Vo = CellVariable(mesh=dim, name = "Vo", value = Cv)
Vr = CellVariable(mesh=dim, name = "Vr", value = 0.0)
Dr = CellVariable(mesh=dim, name = "Dr", value = Cd)
Dg = CellVariable(mesh=dim, name = "Dg", value = 0.0)
P = CellVariable(mesh=dim, name = "P", value = Cp)
Pr = CellVariable(mesh=dim, name = "Pr", value = 0.0)
simTime = Variable()
simTime.setValue(0.0)

# make a non uniform enzyme concentration
myValues = npy.zeros((nx))

if proteinProfile == "40K":
    para = [0.509402219, 62.24455856, 0.042981579, 8.393295183]  # 40K beads
myValues = (para[0] * np.exp(para[1] * (mesh.x - 1.0)) + para[2] * np.exp(para[3] * (mesh.x - 1.0)))

# fit from the albumin data

elif proteinProfile == "original":
    para = [0.423650344, 58.40049825, 0.0966751, 6.349372997]  # original
    myValues = (para[0] * np.exp(para[1] * (mesh.x - 1.0)) + para[2] * np.exp(para[3] * (mesh.x - 1.0)))

# fit from the albumin data

elif proteinProfile == "noVA":
    para = [5.8780046, 0.714285714, 2.819250868]  # noVA
    mask = np.asarray(mesh.x) > para[1]
    myValues[mask] = (para[0] * np.power(mesh.x[mask] - para[1], para[2]) # fit from the no VA data

elif proteinProfile == "excluded":
    mask = np.asarray(mesh.x) > 0.98  # excluded from the bead
    myValues[mask] = 0.5

else:
    printproteinProfile, "is not a valid enzyme profile"
    quit()

print "Enzyme profile set as {}
".format(proteinProfile)

# set intial values note this assumes the initial equilibrium is achieved
Ei.setValue(li * myValues)
Eii.setValue(lii * myValues)
Eiis.setValue(liis * myValues)
Eo.setValue(lo * myValues)
Vr.setValue(Vri)
Pr.setValue(Pri)

# set the surface bc (bc only needed for the species that are moving)
Vo.constrain(Cv + simTime * dvdt, mesh.facesRight)
Vr.constrain(Vri, mesh.facesRight)
P.constrain(Cp + simTime * dpdt, mesh.facesRight)
Pr.constrain(Pri, mesh.facesRight)

# make the center zero flux
Vo.faceGrad.constrain(0.0, mesh.facesLeft)
Vr.faceGrad.constrain(0.0, mesh.facesLeft)
P.faceGrad.constrain(0.0, mesh.facesLeft)

# define a viewer if turned on
if myViewer:
    viewer = Viewer(vars=(Vo, Dg, Dr), datamin=0.0, datamax=Tm1 * Cv * Ce / Tm5)

# define the equations
eq1 = TransientTerm(var=Ei) == Tm1 * Eo * P - (Tm2 + Tm4) * Ei * Vo - Tm3 * Ei * (Dr + Dg / dyeR)
# dEi/dt
eq2 = TransientTerm(var=Eii) == Tm3 * Ei * (Dr + Dg / dyeR) + Tm4 * Ei * Vo - Tm6 * Eii * Vo
# dEii/dt
eq3 = TransientTerm(var=Eiis) == Tm2 * Ei * Vo - Tm5 * Eii
# dEiis/dt
eq4 = TransientTerm(var=Eo) == -Tm1 * Eo * P + Tm5 * Eii + Tm6 * Eii * Vo
# dEo/dt
eq5 = TransientTerm(var=Vo) == DiffusionTerm(coeff=Dvo, var=Vo) - (Tm2 + Tm4) * Ei * Vo - Tm6 * Eii * Vo
# dVo/dt
eq6 = TransientTerm(var=Vr) == DiffusionTerm(coeff=Dvr, var=Vr) - Tm9 * (Dr + Dg / dyeR) * Vr + Tm4 * Ei * Vo + Tm6 * Eii * Vo - Tm7 * Vr / dVr/dt
# dVr/dt
eq7 = TransientTerm(var=Dr) == -(Tm5 * Ei + Tm9 * (Vr + Pr) * Vr)
# dDr/dt
eq8 = TransientTerm(var=Dg) == -(Dr - Dg / dyeR) * Ei + Tm9 * (Vr + Pr)
# dDg/dt
eq9 = TransientTerm(var=P) == DiffusionTerm(coeff=1.0, var=P) - Tm1 * Eo * P + Tm8 * Pr * Pr
# dP/dt
eq10 = TransientTerm(var=Pr) == DiffusionTerm(coeff=1.0, var=Pr) - Tm9 * (Dr + Dg / dyeR) * Pr + Tm7 * Vr - Tm8 * Pr * Pr
# dPr/dt

eq = eq1 & eq2 & eq3 & eq4 & eq5 & eq6 & eq7 & eq8 & eq9 & eq10
myTime = time() # get the initial time

# solve the equations
for step in range(steps):
    try:
        simTime.setValue(simTime+dt)
        res = eq.sweep(dt = dt)
        if myViewer:
            viewer.plot()
    except KeyboardInterrupt:  # this allows me to trap the stop
        break

    if step == 4000:  # bump up the time step after things have settled down
        dt *= 2
    if step % printInterval == 0:
        print "current step = {0:06d}".format(step),
        print "clock time = {:.2f}".format(time()-myTime)
        storeResults()

    if Cp+simTime*dpdt <= 0.0: #stop if all the peroxide is gone.
        break

print "total time simulated = ",{simTime*R**2/Dp
print "elapsed time = ",{time()-myTime, " seconds"
storeResults()

# function to generate a square distance matrix
def distArray(myCount):
    a = npy.linspace(-myCount, myCount, 2*myCount+1)
    b = npy.array([a*a,]*(2*myCount+1))
    return npy.sqrt(b + b.transpose())

# make an array to hold the image
im = npy.zeros((637,637,3), dtype='uint8')
dArray = distArray(318)
dArray /= 318.0

# scale the concentration data
redArray = 255*Dr/Cd
greenArray = 255*Dg/Cd

# make and display the image
im[:,:,0] = interp(dArray,mesh.x,redArray, left = redArray[0], right = 0.0)
im[:,:,1] = interp(dArray,mesh.x,greenArray, left = greenArray[0], right = 0.0)
imLarge = npy.zeros((1024,1024,3), dtype='uint8')
imLarge[194:831,194:831,:] = im

myImage = Image.fromarray(imLarge)
myImage.save("model_"+suffix+".tif")
myImage.show()
# Reaction of HRP, Peroxide and TMB in the beads

## unsteady-state diffusion in a sphere

The species:
- Eo = reduced enzyme
- EI = enzyme I
- EII = enzyme II
- EII = enzyme II associated with substrate
- Vo = veratryl alcohol
- Vr = oxidized veratryl alcohol (cation radical)
- Va = veratryl aldehyde
- Dr = reduced dye (red)
- P = peroxide
- Pr = super oxide

ch 5/18/15

import the modules
from fipy import CellVariable, DiffusionTerm, Viewer, TransientTerm, Variable
from fipy.meshes.sphericalUniformGrid1D import SphericalUniformGrid1D
from time import time
import numpy as npy
from scipy import interp
from PIL import Image

define the parameters
Dp = 5.0E-6          # peroxide diffusion coefficent (cm^2/sec)
R = 35.0E-4          # radius (cm)
Ce = 0.00031746     # enzyme concentration (mM)
Cd = 0.4             # dye concentration (mM)
Cv = 0.000714        # TMB concentration (mM)
Cv *= 2.0            # of doses
Cp = 0.286           # peroxide concentration (mM)
K1 = 580.0           # rxn Eo + P --> EI + H2O (1/mM s) (tien)
K2 = 0.01            # rxn EI + Vo --> EII (1/mM s)
K3 = 1200.0          # rxn EI + D --> EII + Dox (1/mM s)
K4 = 1000.0          # rxn EI + Vo --> EII + Vr (1/mM s)
K5 = 0.01            # rxn EII --> Eo + Va (1/s)
K6 = 396.0           # rxn EII + Vo --> Eo + Vr (1/mM s) (fit)
K7 = 0.0295448258748605 # rxn Vr --> + O2 --> Va + HOO. (1/s) (fit)
K8 = 1000000.0       # rxn HOO. + HOO. --> P + O2 (1/mM s)
K9 = 500.0           # rxn radical + D --> + Dox (1/mM s) (WAG)
dyeR = 1000.0        # ratio of red to green bodipy reaction rate with radical
Dvr = 0.62           # ratio of TMB diffusion to H2O2
Dvo = 0.62           # ratio of TMB cation radical diffusion to H2O2
proteinProfile = "noVA" # set the profile "orginal", "40K", "noVA" or "excluded"

solution parameters
myViewer = False     # if True a plot of concentration vs position is shown and updated
steps = 400000       # maximum number of time steps
totalTime = 200.0    # maximum time (this is the real limit)
suffix = "HRP_2_500"
printInterval = 1000 # number of step between prints
nx = 100             # number of nodes
dx = 1.0 / nx
dt = 0.002

def a functions to store the dye results
def storeResults():
    try:
        myFile = open("model_"+suffix+".csv","w")
    except IOError as err:
        print("can't open file")
    else:
        # put a header on the file
        myFile.write("total time simulated = {} seconds\n".format(simTime*R**2/Dp))
        myFile.write("peroxide diffusion coefficent (cm^2/sec) = {}\n".format(Dp))
myFile.write("radius (cm) = {}\n\nformat(R))
myFile.write("enzyme concentration (mM) = {}\n\nformat(Ce))
myFile.write("peroxide concentration (mM) = {}\n\nformat(Cp))
myFile.write("dye concentration (mM) = {}\n\nformat(Cd))
myFile.write("K1 = {} rxn Eo + P --> Ei + H2O (1/mM s)\n\nformat(K1))
myFile.write("K2 = {} rxn Ei + Vo --> Eiis  (1/mM s)\n\nformat(K2))
myFile.write("K3 = {} rxn Ei + D --> Eiis + Dox (1/mM s)\n\nformat(K3))
myFile.write("K4 = {} rxn Ei + Vo --> Eii + Vr (1/mM s)\n\nformat(K4))
myFile.write("K5 = {} rxn Eiis --> Eo + Va (1/s)\n\nformat(K5))
myFile.write("K6 = {} rxn Eii + Vo --> Eo + Vr (1/mM s)\n\nformat(K6))
myFile.write("K7 = {} rxn Vo + O2 --> Va + HOO. (1/s) (assume O2 is\n\nconstant)\n\nformat(K7))
myFile.write("K8 = {} rxn HOO. + HOO. --> P + O2 (1/mM s)\n\nformat(K8))
myFile.write("enzyme profile = {} updated\n\nformat(proteinProfile))
myFile.write("Position, Ei, Eii, Eiis, Eo, Dye(red), Dye(green), TMB, Cation\n\nRadical, Peroxide, Hydroperoxyl Radical\n")
for j in range(0, nx):
    myFile.write("{}\n\nformat(mesh.x[j], Ei[j], Eii[j], Eiis[j], Eo[j], Dr[j], Dg[j], Vo[j], Vr[j], P[j], Pr[j]))
myFile.close()

# dimensionless rate constants
Tm1 = K1*R**2/Dp
Tm2 = K2*R**2/Dp
Tm3 = K3*R**2/Dp
Tm4 = K4*R**2/Dp
Tm5 = K5*R**2/Dp
Tm6 = K6*R**2/Dp
Tm7 = K7*R**2/Dp
Tm8 = K8*R**2/Dp
Tm9 = K9*R**2/Dp

# rate of veratyl alcohol and peroxide loss
li = Ce/(1+Tm2*Cv/Tm5 + Tm4/Tm6 + (Tm2+Tm4)*Cv/(Tm1*Cp)) # initial enzyme conc.
lii = Tm4*li/Tm6
liis = Tm2*li*Cv/Tm5
lo = (Tm2+Tm4)*li*Cv/(Tm1*Cp)
VoK = ((Tm2 + Tm4)*lii + Tm6*liii) # rate of change of TMB in solution (assume P excess)
VrK = Cv*Tm6*Ce/(Tm7-Tm6*Ce)
PrK = VrK*Tm7/Tm8

# build a 1D mesh
mesh = SphericalUniformGrid1D(nx=nx, dx=dx)

# initialize the cell variables
Eo = CellVariable(mesh=mesh, name = "Eo", value = 0.0) # this is set below
Ei = CellVariable(mesh=mesh, name = "Ei", value = 0.0) # this is set below
Eiis = CellVariable(mesh=mesh, name = "Eiis", value = 0.0) # this is set below
Eii = CellVariable(mesh=mesh, name = "Eii", value = 0.0) # this is set below
Vo = CellVariable(mesh=mesh, name = "Vo", value = Cv)
Vr = CellVariable(mesh=mesh, name = "Vr", value = 0.0)
Dr = CellVariable(mesh=mesh, name = "Dr", value = Cd)
Dg = CellVariable(mesh=mesh, name = "Dg", value = 0.0)
P = CellVariable(mesh=mesh, name = "P", value = Cp)
Pr = CellVariable(mesh=mesh, name = "Pr", value = 0.0)
simTime = Variable()
simTime.setValue(0.0)

# make a non uniform enzyme concentration
myValues = nyp.zeros((nx))
if proteinProfile == "40K":
    para = [0.509402219, 62.24455856, 0.042981579, 8.393295183] # 40K beads
myValues = (para[0]*npy.exp(para[1]*(mesh.x-1.0)) + para[2]*npy.exp(para[3]*(mesh.x-1.0)))

# fit from the albumin data
if proteinProfile == "original":
    para = [0.423650344, 58.40049825, 0.0966751, 6.349372997]  # original
    myValues = (para[0]*npy.exp(para[1]*(mesh.x-1.0)) + para[2]*npy.exp(para[3]*(mesh.x-1.0)))
    # fit from the albumin data
elif proteinProfile == "noVA":
    para = [5.8780046, 0.714285714, 2.819250868]  # noVA
    mask = npy.asarray(mesh.x) > para[1]
    myValues[mask] = (para[0] * npy.power(mesh.x[mask] - para[1], para[2]))  # fit from the no VA data
elif proteinProfile == "excluded":
    mask = npy.asarray(mesh.x) > 0.98  # excluded from the bead
    myValues[mask] = 0.5
else:
    print proteinProfile, "is not a valid enzyme profile"
    quit()

print "Enzyme profile set as {}\n".format(proteinProfile)

# set initial values note this assumes the initial equilibrium is achieved
Ei.setValue(li*myValues)
Eii.setValue(lii*myValues)
Eiis.setValue(liis*myValues)
Eo.setValue(lo*myValues)

# set the surface bc (bc only needed for the species that are moving)
Vo.constrain(Cv*npy.exp(-VoK*simTime), mesh.facesRight)
Vr.constrain(VrK*(npy.exp(-Tm6*Ce*simTime)-npy.exp(-Tm7*simTime)), mesh.facesRight)
P.constrain(Cp, mesh.facesRight)
Pr.constrain(npy.sqrt(PrK*(npy.exp(-Tm6*Ce*simTime)-npy.exp(-Tm7*simTime))), mesh.facesRight)

# make the center zero flux
Vo.faceGrad.constrain(0.0, mesh.facesLeft)
Vr.faceGrad.constrain(0.0, mesh.facesLeft)
P.faceGrad.constrain(0.0, mesh.facesLeft)
Pr.faceGrad.constrain(0.0, mesh.facesLeft)

# define a viewer if turned on
if myViewer:
    viewer = Viewer(vars=(Vo, Dg, Dr), datamin=0.0, datamax=Tm1*Cv*Ce/Tm5)

# define the equations
eq1 = TransientTerm(var = Ei) == Tm1*Eo*P - (Tm2+Tm4)*Ei*Vo - Tm3*Ei*(Dr + Dg/dyeR)
# dEi/dt
eq2 = TransientTerm(var = Eii) == Tm3*Ei*(Dr + Dg/dyeR) + Tm4*Ei*Vo - Tm6*Eii*Vo
# dEii/dt
eq3 = TransientTerm(var = Eiis) == Tm2*Ei*Vo - Tm5*Eii
# dEiis/dt
eq4 = TransientTerm(var = Eo) == -Tm1*Eo*P + Tm5*Eiiis + Tm6*Eii*Vo
# dEo/dt
eq5 = TransientTerm(var = Vo) == DiffusionTerm(coef = Dvo, var = Vo) - (Tm2+Tm4)*Ei*Vo - Tm6*Eii*Vo
# dVo/dt
eq6 = TransientTerm(var = Vr) == DiffusionTerm(coef = Dvr, var = Vr) - Tm9*(Dr + Dg/dyeR)*Vr + Tm4*Ei*Vo + Tm6*Eii*Vo - Tm7*Vr
# dVr/dt
eq7 = TransientTerm(var = Dr) == -Tm3*Dr*Ei - Tm9*Dr*(Vr+Pr)
# dDr/dt
eq8 = TransientTerm(var = Dg) == (Dr - Dg/dyeR)*(Tm3*Ei + Tm9*(Vr+Pr))
# dDg/dt
eq9 = TransientTerm(var = P) == DiffusionTerm(coef = 1.0, var = P) - Tm1*Eo*P + Tm8*Pr*Pr
# dP/dt
eq10 = TransientTerm(var = Pr) == DiffusionTerm(coef = 1.0, var = Pr) - Tm9*(Dr + Dg/dyeR)*Pr + Tm7*Vr - Tm8*Pr*Pr
# dPr/dt

eq = eq1 & eq2 & eq3 & eq4 & eq5 & eq6 & eq7 & eq8 & eq9 & eq10

myTime = time()  # get the initial time
# solve the equations
for step in range(steps):
    try:
        simTime.setValue(simTime+dt)
        res = eq.sweep(dt = dt)
        if myViewer:
            viewer.plot()

        if step == 1000:  # bump up the time step after things have settled down
            dt *= 2

        if step % printInterval == 0:
            print "current step = {0:06d}".format(step),
            print "clock time = {:.2f}".format(time()-myTime)
            storeResults()

        if totalTime <= simTime*R**2/Dp: #stop at a set time
            break

    except KeyboardInterrupt:  # this allows me to trap the stop
        break

print "total time simulated = ", simTime*R**2/Dp
print "elapsed time = ", time()-myTime, " seconds"
storeResults()

def distArray(myCount):
    a = npy.linspace(-myCount, myCount, 2*myCount+1)
    b = npy.array([a*a,]*(2*myCount+1))
    return npy.sqrt(b + b.transpose())

def make an array to hold the image
im = npy.zeros((637,637,3), dtype='uint8')
dArray = distArray(318)
dArray /= 318.0

# scale the concentration data
redArray = 255*Dr/Cd
greenArray = 255*Dg/Cd

# make and display the image
im[:,:,0] = interp(dArray,mesh.x,redArray, left = redArray[0], right = 0.0)
im[:,:,1] = interp(dArray,mesh.x,greenArray, left = greenArray[0], right = 0.0)
imLarge = npy.zeros((1024,1024,3), dtype='uint8')
imLarge[194:831,194:831,:] = im

myImage = Image.fromarray(imLarge)
myImage.save("model_"+suffix+".tif")
myImage.show("model_"+suffix+".tif")
# Code for doing image analysis on beads

```python
import cv2
import numpy as np
from matplotlib import pyplot as plt
from scipy import misc
from PIL import Image

# function to generate a square distance matrix
def distArray(myCount):
    a = np.linspace(-myCount, myCount, 2*myCount+1)
    b = np.array([a*a,]*(2*myCount+1))
    return np.sqrt(b + b.transpose())

# file data [name, green power, red power, pixel size, image size]
file = [
    ['01_686P38_A', 100.0, 70.0, 0.11, 1024],  # 0
    ['02_686P38_A', 100.0, 80.0, 0.11, 1024],  # 1
    ['03_686P38_A', 100.0, 90.0, 0.11, 1024],  # 2
    ['04_686P38_A', 100.0, 85.0, 0.11, 1024],  # 3
    ['05_686P38_A', 100.0, 100.0, 0.11, 1024],  # 4
    ['06_686P38_A', 100.0, 70.0, 0.11, 1024],  # 5
    ['07_686P38_A', 100.0, 65.0, 0.11, 1024],  # 6
    ['08_686P38_A', 100.0, 65.0, 0.11, 1024],  # 7
    ['09_686P38_A', 60.0, 40.0, 0.11, 1024],  # 8
    ['10_686P38_A', 70.0, 50.0, 0.11, 1024],  # 9
    ['11_686P38_A', 100.0, 55.0, 0.11, 1024],  # 10
    ['12_686P38_B', 100.0, 65.0, 0.11, 1024],  # 11
    ['13_686P38_B', 100.0, 60.0, 0.11, 1024],  # 12
    ['14_686P38_B', 100.0, 55.0, 0.11, 1024],  # 13
    ['15_686P38_B', 100.0, 60.0, 0.11, 1024],  # 14
    ['16_686P38_B', 100.0, 62.0, 0.11, 1024],  # 15
    ['17_686P38_B', 100.0, 55.0, 0.11, 1024],  # 16
    ['18_686P38_B', 100.0, 47.0, 0.11, 1024],  # 17
    ['19_686P38_B', 100.0, 50.0, 0.11, 1024],  # 18
    ['20_686P38_B', 100.0, 60.0, 0.11, 1024],  # 19
    ['21_686P38_B', 100.0, 70.0, 0.11, 1024],  # 20
    ['22_686P38_C', 100.0, 85.0, 0.11, 1024],  # 21
    ['23_686P38_C', 100.0, 60.0, 0.11, 1024],  # 22
    ['24_686P38_C', 100.0, 70.0, 0.11, 1024],  # 23
    ['25_686P38_C', 100.0, 80.0, 0.11, 1024],  # 24
    ['26_686P38_C', 100.0, 75.0, 0.11, 1024],  # 25
    ['27_686P38_C', 100.0, 90.0, 0.11, 1024],  # 26
    ['28_686P38_C', 100.0, 100.0, 0.11, 1024],  # 27
    ['29_686P38_C', 100.0, 50.0, 0.11, 1024],  # 28
    ['30_686P38_C', 100.0, 72.0, 0.11, 1024],  # 29
    ['31_686P38_C', 100.0, 68.0, 0.11, 1024],  # 30
    ['32_686P38_C', 100.0, 60.0, 0.11, 1024],  # 31
    ['33_686P38_C', 100.0, 55.0, 0.11, 1024],  # 32
    ['34_686P38_D', 100.0, 65.0, 0.11, 1024],  # 33
    ['35_686P38_D', 100.0, 62.0, 0.11, 1024],  # 34
    ['36_686P38_D', 100.0, 60.0, 0.11, 1024],  # 35
    ['37_686P38_D', 100.0, 65.0, 0.11, 1024],  # 36
    ['38_686P38_D', 100.0, 50.0, 0.11, 1024],  # 37
    ['39_686P38_D', 100.0, 55.0, 0.11, 1024],  # 38
    ['40_686P38_D', 100.0, 40.0, 0.11, 1024],  # 39
    ['41_686P38_D', 100.0, 50.0, 0.11, 1024],  # 40
    ['42_686P38_D', 100.0, 53.0, 0.11, 1024],  # 41
    ['43_686P38_E', 100.0, 35.0, 0.11, 1024],  # 42
    ['44_686P38_E', 100.0, 25.0, 0.11, 1024],  # 43
    ['45_686P38_E', 100.0, 30.0, 0.11, 1024],  # 44
    ['46_686P38_E', 90.0, 20.0, 0.11, 1024],  # 45
    ['47_686P38_E', 90.0, 12.0, 0.11, 1024],  # 46
    ['48_686P38_E', 100.0, 30.0, 0.11, 1024],  # 47
    ['49_686P38_E', 100.0, 35.0, 0.11, 1024],  # 48
    ['50_686P38_E', 100.0, 42.0, 0.11, 1024],  # 49
    ['51_686P38_E', 100.0, 30.0, 0.11, 1024],  # 50
    ['52_686P38_E', 100.0, 40.0, 0.11, 1024],  # 51
    ['53_686P38_BODIPY40k', 100.0, 40.0, 0.11, 1024],  # 52
    ['54_686P38_BODIPY40k', 100.0, 45.0, 0.11, 1024]  # 53
]
```
i = 0  # change this to select among the files

# calculate a few parameters for the analysis
pixelSize = file[i][3]
greenScale = 2.0*file[i][2]/file[i][1]
threshold = 40

# generate contours from a gray-scale image
im1 = cv2.imread(file[i][0] + '.tif')
if im1 == None:
    print file[i][0], "not found"
    exit()

imgray = cv2.cvtColor(im1,cv2.COLOR_BGR2GRAY)
ret1,thresh1 = cv2.threshold(imgray,threshold,255,cv2.THRESH_BINARY)
contours1, hierarchy1 = cv2.findContours(thresh1,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)

# get rid of any tiny regions
shortcnt1 = []
for cnt in contours1:
    if (cv2.contourArea(cnt)>2000):
        shortcnt1.append(cnt)

# process the contours
myIndex = 1
for cnt in shortcnt1:

    # calculate center and radius
    M = cv2.moments(cnt)
    cx = int(M['m10']/M['m00'])
    cy = int(M['m01']/M['m00'])
    cr = int(np.sqrt(cv2.contourArea(cnt)/np.pi))
    cp = int(cv2.arcLength(cnt,True)/(2.0*np.pi))

    print myIndex,cx,cy,cr,float(cp)/float(cr)

    # reopen the image
    im = Image.open(file[i][0] + '.tif')

    # split the bands
    R, G, B = 0, 1, 2
    bands = im.split()

    # make arrays
    imarrayR = np.array(bands[R])
    imarrayG = np.array(bands[G])
    imarrayB = np.array(bands[B])

    # only do if the whole bead is in the image and the contour is smooth and round
    if (cy-cr > 0 ) and (cy+cr < file[i][4]) and (cx-cr > 0 ) and (cx+cr < file[i][4]) and (float(cp)/float(cr) < 5.0):
        print file[i][0]
        print 'center (x,y) =',cr,cy, 'radius =',cr, 'circularity = ',float(cp)/float(cr)

        # extract the data
        myDataArrayR = imarrayR[cy-cr:cy+cr+1,cx-cr:cx+cr+1].astype(float)
        myDataArrayG = imarrayG[cy-cr:cy+cr+1,cx-cr:cx+cr+1].astype(float)
        myDistArray = distArray(cr)
        myDistArray *= pixelSize  # pixel size correction
        rMax = (cr+1)*pixelSize
#use the nonbead area to get the background intensity
outerMask = myDistArray > rMax + 2.5
outerMask *= myDataArrayR < threshold
outerMask *= myDataArrayG < threshold

dataRed = np.sum(outerMask*myDataArrayR).astype(float)
dataGrn = np.sum(outerMask*myDataArrayG).astype(float)
dataNum = np.sum(outerMask).astype(float)

minRed = dataRed/dataNum
minGrn = dataGrn/dataNum

print 'backgrounds (red, green) =', minRed, minGrn

#subtract the camera offset
myDataArrayR -= int(minRed)
myDataArrayG -= int(minGrn)

#mask to remove off scale (i.e. 255 - offset) unsigned ints wrap around
myMask = myDataArrayG < (255 - int(minGrn))
myMask *= myDataArrayR < (255 - int(minRed))

#apply the mask to the arrays
myDataArrayR *= myMask
myDataArrayG *= myMask

#generate the annular data
numSlice = 100
dataD = np.zeros(numSlice+1)
dataR = np.zeros(numSlice+1)
dataG = np.zeros(numSlice+1)
dataN = np.zeros(numSlice+1)
distance = np.linspace(0.0, rMax, num=numSlice, endpoint=True)
incr = (distance[1] - distance[0])/2

index = 0
for r in distance:
    ringMask = myDistArray > r
    ringMask *= myDistArray <= r + 1.0
    dataD[index] = r + incr*pixelSize
    dataR[index] = np.sum(ringMask*myDataArrayR)
    dataG[index] = np.sum(ringMask*myDataArrayG)
    dataN[index] = np.sum(ringMask)
    index += 1

#calculate the fraction green
dataG *= greenScale

dataTotal = dataR + dataG

dataTotal += dataTotal == 0.0 # eliminate any zero values
dataRatio = dataG / (dataTotal)

# Make the xy plot
fig = plt.figure()
ax = fig.add_subplot(111)

# Add title, and axis title
ax.set_title("ratio vs distance")
ax.set_xlabel("distance (um)")
ax.set_xlim(0, np.max(distance))
ax.set_ylim(0, 1)

# Add my data
ax = plt.scatter(dataD, dataRatio, c='k')
plt.show()

myIndex += 1

#store the ratio histogram
try:
    myFile = open(file[i][0] + ".csv","w")
except IOError as err:
    print("can't open file")
else:
    for j in range(0, numSlice):
        myFile.write("{},{},{}
.format(dataD[j], dataRatio[j], dataN[j]))
myFile.close()

# draw a circle to be sure we got it correct
im1 = np.array(im)
cv2.circle(im1, (cx,cy), cr, (255,255,255), 2, 8)
   cv2.drawContours(im1, cnt, -1, (255,255,255), 1)
   Image.fromarray(im1).save('confocal_cal.tif')
Image.fromarray(im1).show()
Fungal lignin peroxidase does not produce the veratryl alcohol cation radical as a diffusible ligninolytic oxidant
Carl J. Houtman, Eranda Maligaspe, Christopher G. Hunt, Elena Fernandez-Fueyo, Angel T. Martínez and Kenneth E. Hammel

*J. Biol. Chem.* published online February 9, 2018

Access the most updated version of this article at doi: [10.1074/jbc.RA117.001153](http://10.1074/jbc.RA117.001153)

Alerts:
- When this article is cited
- When a correction for this article is posted

Click here to choose from all of JBC's e-mail alerts